

# VIRTUAALITODELLISUUSPELIN MALLINTAMINEN

Tuominen Arli

Opinnäytetyö  
Kulttuuriala  
Kuvataide  
Kuvataiteilija (AMK)

2017

Kulttuuriala  
Kuvataide  
Kuvataiteilija (AMK)

---

<b>Tekijä</b>	Arli Tuominen	<b>Vuosi</b>	2017
<b>Ohjaaja(t)</b>	Jari Penttinen		
<b>Toimeksiantaja</b>			
<b>Työn nimi</b>	Virtuaalitodellisuuspelin mallintaminen		
<b>Sivu- ja liitesivumäärä</b>	47 + 0		

---

Opinnäytetyössäni tutkin, mitä visualistin pitää ottaa huomioon, kun visuaalista sisältöä tuotetaan virtuaalitodellisuuspeliin. Opinnäytteeni aihe tuli työharjoittelussa vaihdon aikana Ulsterin yliopistossa, jossa olin taiteellisenä johtajana. Tuona aikana tein kaiken visuaalisen sisällön virtuaalitodellisuuspeliin. Opinnäytetyöni tarkoitus on antaa vankka pohja seuraaville visuaalisen sisällön tuottajille virtuaalitodellisuusteknologiaa varten. Tutkimustyötäni voin hyödyntää tulevassa ammatissani peliteollisuuden visuaalisella puolella.

Tutkin, mitä virtuaalitodellisuudella tarkoitetaan ja miten 3D-mallinnuksen prosessi etenee. Kerron mallintamisprosessin yksityiskohtaisesti. Esittelen eri ohjelmien eroavaisuuksia ja ominaisuuksia. Esittelen eri virtuaalitodellisuuden käyttökohteita, tapoja sekä historiaa. Näissä tutkimuksissa hyödynsin eritoten lähteinä alan ammattilaisten verkkosivuja sekä hieman myös alan kirjallisuutta. Uuden teknologian takia aiheesta ei vielä löydy paljon kirjallisuutta.

Käyn läpi vaihe vaiheelta visuaalisen sisällön tuottamisprosessin virtuaalitodellisuuspeliin. Keskityn jokaisen vaiheen luomisprosessiin ja siihen, kuinka sain vaiheen toimimaan virtuaalitodellisuudessa. Kerron tutkimukseni aikana ilmeneistä ongelmista ja niiden ratkaisuista.

Tutkimuksessani käytin 3D-mallintamiseen Autodeskin 3Ds Max -ohjelmistoa, pelin luomiseen Unity-pelimootoria, tekstuurien luomiseen Adoben Photoshop-piirustusohjelmaa ja pelin logon tekemiseen käytin Inkscapen vektoripohjaista piirustusohjelmaa. Erikoistekstuurien tekemisen apuna käytin myös Substancen BitMap2Material -mappien luomisohjelmaa. Virtuaalitodellisuuslaitteistona käytin Valven HTC Viveä.

Avainsanat

3D-mallinnus, virtuaalitodellisuus, 3Ds Max, Unity, HTC Vive, teksturointi, visualisti

Lapland University of Applied Sciences  
Bachelor of Culture and Arts

---

---

<b>Author</b>	Arli Tuominen	<b>Year</b>	2017
<b>Supervisor</b>	Jari Penttinen		
<b>Commissioned by</b>			
<b>Subject of thesis</b>	Modeling virtual reality game		
<b>Number of pages</b>	47 + 0		

---

In my thesis I examine what visual designer should take into account when creating visual content for virtual reality game. The subject of my thesis came from my mobile traineeship in Ulster University, where I worked as the lead artist. During the traineeship I created whole visual content for virtual reality game. The purpose of my thesis is to give solid foundation for next visual content creators in virtual reality technology. This research is helpful in my future career as an artist in game industry.

I'm researching what does virtual reality mean and how the process of 3D-modeling progress. I'm explaining the 3D-modeling in its detail. I'm introducing different programs while listing their features and comparing differences. I examine different types and uses of virtual reality. I describe the historical aspect of virtual reality. For research I used source materials such as websites and literature written by the professionals in the field. In the field of virtual reality there are currently only a few literature sources available.

I'm introducing entire process of creating visual content for virtual reality game. I'm explaining the creative process of games visual designing and its use in virtual reality technology. I'm also describing the problems and difficulties which I've encountered during this research, to which I offer solutions.

In my research I used Autodesk's 3Ds Max for creating 3D-models, Unity game engine for creating the game, Adobe's Photoshop drawing program for texturing and vector based drawing program Inkscape for creating the game logo. For special textures I used Substance's Bitmap2Material map creating program. This project was developed for Valve's virtual reality device HTC Vive.

**Key words** 3D-modeling, virtual reality, 3Ds Max, Unity, HTC Vive, texturing, visual design

## SISÄLLYS

1 JOHDANTO.....	6
2 3D-MALLINTAMISEN TYÖVAIHEET.....	8
2.1 3D-mallinnus.....	8
2.2 Teksturointi.....	8
2.3 Rigging.....	9
2.4 Animointi.....	10
2.5 Eksportointi.....	11
3 VIRTUAALITODELLISUUS JA KEHITYSTYÖKALUT.....	13
3.1 Määritelmä.....	13
3.2 Virtuaalitodellisuuden historia.....	13
3.3 Ohjelmistot ja laitteet.....	17
4 VIRTUAALITODELLISUUSPELIN VISUALISOINTI.....	19
5 JOHTOPÄÄTÖKSET.....	45
6 POHDINTA.....	47
LÄHTEET.....	48

## KÄYTETYT MERKIT, LYHENTEET JA VIERASSANAT

asset	kolmannen osapuolen tuottamaa materiaalia pelimootorin käyttöön (Unity 2017b)
VR	virtual reality eli virtuaalitodellisuus
map	bittikarttakuvatyyppe
polygoni	kaksiulotteinen monikulmio 3D-mallissa (Silverman 2013)
renderointi	(tai renderöinti) reaaliaikainen kuvanluominen 3D-mallista, joka sisältää tiedon mm. geometriasta, tekstuurista, valaistuksesta, efekteistä ja katselukulmasta (Birn 2002)

## 1 JOHDANTO

Opinnäytteessäni tutkin, mitä visualistin pitää ottaa huomioon kun, luodaan mallintamalla visuaalista sisältöä virtuaalitodellisuuspeliin. Opinnäytetyöni aihe tuli työharjoittelujaksoltani, jossa tein virtuaalitodellisuuspelin visualisoinnit Ulsterin yliopistossa Irlannissa. Virtuaalitodellisuus eli VR on viimeisimpiä peliteknologian aloja, ja uutena teknologiana se tuo paljon erilaisia haasteita visualistin työhön. Suurimpia vaikeuksia uudessa teknologiassa on se, ettei tutkittua tietoa parhaista ratkaisusta ole vielä saatavilla, vaan ne on tällä hetkellä selvitettävä itse. Juuri tämän opinnäytteeni tekee ja antaa vastauksia sekä vinkkejä niille, jotka pyrkivät ymmärtämään tai itse luomaan VR:n visuaalista puolta. Opinnäytteeni rajoittuukin vain visuaaliseen sisällön tuottamiseen ja eritoten 3D-mallintamiseen ja kaikkiin siihen kuuluviin työvaiheisiin. Pelin visuaaliseen sisältöön kuuluu lyhyesti myös pelissä näkyvän logon suunnittelu ja toteutus. Virtuaalitodellisuutta käytetään myös simulaatioissa ja muissa erilaisissa tekniikoissa, jotka käyn lyhyesti läpi, mutta opinnäytteeni rajoittuu vain pelien virtuaalitodellisuuteen. Tutkimukseni myös hipaisee pelimoottorien ja itse virtuaalitodellisuuslaitteiden toimintaa, visualistin näkökulmasta. VR-laitteiden valmistajia nimenomaan peleihin on tällä hetkellä kolme päätekijää: Facebook:n Oculus Rift, Sony Playstation VR ja Valve:n HTC Vive. Opinnäytteeni keskittyy HTC Vive:n ympärille, mutta myös muut tekijät otetaan myös yleisesti huomioon.

Pelin visuaaliseen suunnitteluun kuuluu muun muassa suunnittelu ja luonnosprosessi, mallintaminen, teksturointi, animointi ja kenttäsuunnittelu. Näihin sisältyy myös kaikki pelin elementit sekä luovaan tekemiseen kuuluvan suunnittelu-prosessin. Työni pyrkii vastaamaan myös siihen, mitä kaikkea pitää ottaa huomioon, kun suunnitellaan visuaalista sisältöä virtuaalitodellisuuspeliin ja mitä mahdollisia rajoitteita tekniikkaan liittyy. Testaan ja tutkin, miten tehdä sekä toimiva että näyttävä VR-peli. Ohjelmistot, joita olen tässä tutkimuksessa pääasiassa käyttänyt, ovat Autodesk:n 3D-mallinnusohjelma 3Ds Max, Adobe:n kuvanmuokkausohjelma Photoshop, vektori pohjainen piirustus ohjelma Inkscape ja pelimoottori Unity.

Opinnäytettä varten tutkimusta sekä peliä tehdessäni virtuaalitodellisuustekniikkaa oli saatavilla vain kehityskäyttöön, eikä laitteita ollut vielä yleisesti myynnissä. Tekniikka on kehittynyt jo paljon eteenpäin siitä, kun opinnäytteeni kirjoitus-

prosessi pääsi alkuun. Kirjoittaessani tutkimuksestani pelejä sekä laitteita oli jo yleisessä myynnissä, myös VR-laitteet olivat kehittyneet jo jonkin verran eteenpäin. Mutta vaikka testaukseni liittyvät VR-pelien ensikosketukseen, moni asia on silti vieläkin ajankohtainen. Samat perusperiaatteet pätevät valmistajasta ja tekniikan kehityksestä riippumatta.

Tekniikka tuo suuria haasteita visuaaliseen suunnitteluun. Kaiken normaalin tekemisen lisäksi myös otettava huomioon pelaajan tasapaino, käytettävä tila, suunta ja hyvinvointi. VR-tekniikkaa käytettäessä väärillä valinnoilla peliä pelaava henkilö voi esimerkiksi kokea pahoinvointia, kaatua tai saada jopa sairauskohtauksen. Näiden syiden takia on erittäin tärkeää, että visuaalinen sisältö on sekä ammattimaisesti että huolellisesti tehty.

Huolelliseen visuaalisen sisällön tuottamiseen kuuluu sen toistuva testaus. Tekemällä usein testejä ja kohtaamalla paljon virheitä, oli mahdollista rajata, mikä toimii virtuaalitodellisuudessa ja mikä ei. Se, mikä näyttää hyvältä pelimoottorin sisällä, ei välttämättä kerro mitään siitä, miten visuaalisen materiaalin kokee VR-lasit päässä. On myös otettava huomioon yksilöiden erot. Mitä enemmän eri ihmisiä kokeilee visualisointia, sitä varmemmin se toimii eri ihmisillä. Tästä kaikesta on mahdollisuus saada selville, mitkä muodot, sävyt tai sommitelmat voivat laukasta huonovointisuutta erityyppisissä ihmisissä. Opinnäytetyöni tutkimuksessa on käytetty useampaa ihmistä, kun näitä kaikkia ominaisuuksia on testattu.

## 2 3D-MALLINTAMISEN TYÖVAIHEET

### 2.1 3D-mallinnus

3D:llä, eli kolmiulotteisella mallinnuksella tarkoitetaan menetelmää, jossa tietokoneen avustuksella ja mallintamiseen erikoistuneella ohjelmalla suunnitellaan kolmiulotteisuutta kuvaruudulla (Wikipedia 2017a,1).

3D-mallintamiseen kuuluu itse mallien suunnittelu ja rakentaminen. Valmis malli koostuu yksityiskohdista ja valmiista muodoista. Mallinnuksen ollessa valmiina se toimii pohjana 3D-mallinnuksen seuraavissa työvaiheissa, kuten teksturoinnissa, luuston lisäämisessä (bones) ja animoinnissa. (Silverman 2013, 1.)

### 2.2 Teksturointi

Teksturoinnilla eli pintakuviointilla tai testuurimappauksella tarkoitetaan tietokonegrafiikassa geometrisen perusmuodon päällystämistä bittikarttakuvalla, eli tekstuurilla. Käytettävä tekstuuri on yleensä staattinen kaksiulotteinen bittikartakuva, mutta tekstuuri voi olla myös kolmiulotteinen tai animoitu. Tekstuuri voi tarkoittaa myös muuta kuin kuvaa, esimerkiksi bump mapilla vaikutetaan valaistuksen parametreihin ja muutetaan 3D-mallin pintastruktuuria. (Wikipedia 2017d, 1.)

Teksturointi mahdollistaa valmiiseen malliin monimutkaisempia pinnanmuotoja niin, ettei itse mallin monimutkaisuutta tarvitse lisätä. Mitä yksinkertaisempaa eli low-polygonina itse mallin pystyy pitämään, sitä nopeampaa on mallin renderointi, joka pelimaailmassa on oleellisen tärkeää. Ammattimaisella teksturoinnilla on mahdollista saada polygoneiltaan yksinkertaisesta mallista näyttävä ja yksityiskohtainen ilman polygonien lisäämistä. (Silverman 2013, 1.)

Teksturointitekniikoita on useita: yksinkertaisesta UVW map -tekniikasta, jossa yksinkertaisilla säätimillä on mahdollista sovittaa valmis tekstuuri mallin päälle, monimutkaiseen unwrap UVW -tekniikkaan, jossa luodaan täysin oma UVW mappaus UVW- editorilla ja kuvanmuokkausohjelmalla. Erona näillä kahdella tekniikalla on se, että yksinkertainen UVW-map pudotetaan 3D-mallin päälle ja sitä siirrellään ja venytetään, kun taas unwrap UVW -tekniikalla 3D-mallin pinta "kuoritaan" litteäksi. Viimeksi mainittu tekniikka mahdollistaa sen, että kaikki nä-



kyvät pinnat on mahdollista muokata tarkasti ja epäsymmetrisesti. (Wang 2016, 23–30.)

### 2.3 Rigging

Riggingillä tarkoitetaan 3D-malliin sijoitettua luustoa ja mallin pinnan manipulaatiota (Slick 2016, 1).

Ennen kuin monimutkaista kolmiulotteista mallia kannattaa animoida, se on ensin ”rigattava”. Rigging-prosessiin kuuluu luuston eli rigid bodyn -luominen, envelopes-editointi sekä mahdollinen ”weight mappaus”. Monimutkainen malli voi olla esimerkiksi ihmis- tai eläinhahmo, malli, jolla on esimerkiksi neljä raajaa. Jotta hahmo ja sen iho kääntyvät ja vääntyvät luontevasti, on sille ensin rakennettava rigid-body eli ”luusto”. 3Ds Max -ohjelmistossa ihmishahmoille on valmiiksi oma rigid body, ja erilaisille eläinhahmoille on ns. CAT-modifierit. Kun 3D-mallissa on valmis riggaus, sen voi animoida suoraan 3D-mallinnusohjelmistossa tai pelimoottorissa. Animoinnin voi tehdä ilman riggausta, mutta tällä tekniikalla säästyy paljon aikaa ja vaivaa, sekä mallin liikkeistä on mahdollista saada paljon luontevampia. (Autodesk 2016, 1; Unity 2017c, 1.)

Kun itse luusto on nivelineen valmis, rigging-prosessin seuraavaa vaihetta kutsutaan nimellä Envelopes editointi. Tällä tarkoitetaan sitä, miten 3D-mallin pinta tai ”iho” käyttäytyy, kun luusto liikkuu. Envelopes-työvaiheessa määritetään kaikki ihon osat, jotka liikkuvat tai venyvät kun esimerkiksi hahmon polvi nousee. (Masters 2014a, 1.)

Kun envelopes editointi -vaihe on suoritettu loppuun, voi ihon muuntumista liikkeessä muuttaa myös Weight mapin avulla. Weight mappauksella määritetään, paljonko tietty ihon osa liikkuu suhteessa ympäröivään ihoon, kun tietty luu liikkuu. Tällä tekniikalla on mahdollista poistaa kaikki ongelmalliset kohdat, jossa mallin iho voi mennä päällekkäin. Tämä voi tapahtua ns. ääriliikkeissä, kuten käsivarren koukistamisessa. Tämän jälkeen rigattu 3D-malli on täysin valmis animaatioprosessiin 3D-mallinnus ohjelmistoissa tai pelimoottorissa. (Freescale 2010, 1.)

Erittäin vaativissa 3D-malleissa on myös mahdollista rigata lihakset ja niiden kaikki ominaisuudet, muodon muutoksesta liikkeen säätelyyn. Lihakset liite-

tään luiden päälle ja lihasten ominaisuuksia muuttamalla ne voivat liikuttaa luita ja myöhemmin myös ihoa erittäin luonnonmukaisesti. Tämä tekniikka on hyödyllinen, kun tavoitteena on anatomisesti tarkka 3D-mallin animaatio. Rigatut lihakset on mahdollista eksportoida myös Unity:n, mutta tämä prosessi vaatii laajaa ohjelmiston osaamista. (Simmons, Wilhelms & Van Gelder 2002, 143.)

## 2.4 Animointi

3D-animoinnilla tarkoitetaan kolmiulotteisen objektin liikkumista virtuaalisessa tilassa (PC 2016, 1).

Yksinkertainen malli, kuten pallo, ei välttämättä tarvitse riggausta ollenkaan, jotta sen voi animoida. 3D-mallinnusohjelmissa, pelimoottorissa tai jopa pelkällä scriptauksella (koodaamisella) mallia voidaan pyörittää ja vääntää ilman, että rigging työvaihe olisi tarpeen. Mutta mitä monimutkaisempi animoitava 3D-malli, sitä hyödyllisemmäksi rigging työvaihe tulee. Esimerkiksi ihmisen 3D-mallin animoiminen on vaikea tehdä näyttävästi ja tehokkaasti ilman rigging-työvaihetta. (Masters 2014b, 1.)

Valmiiksi riggattu 3D-mallin voidaan tuoda pelimoottoriin, jossa animoinnin suorittaa pelimoottorin oma ohjelmisto. Rigid bodyn yksittäiset luut voidaan ohjelmoida liikkumaan tietyllä tavalla, kun eri kontrolleja painetaan. Tässä menetelmässä voidaan myös käyttää osittain valmiiksi animoituja 3D-malleja, joissa eri kontrollien painaminen käynnistää eri animaatioita. Esimerkkinä tästä ovat peleissä käytetyt juoksu- sekä kävelyanimaatiot, jotka asteittain laukeavat kun eri kontrolleista painetaan. (Ehrenhaus 2014, 1.)

Sekä 3D-mallinnusohjelmistoissa että pelimoottoreissa on mahdollista myös lisätä fysiikan piirteitä, joka vaikuttaa mallin liikkeisiin. On mahdollista esimerkiksi luoda painovoima tai vaikka tuulen vaikutus, joka vaikuttaa mallin tapaan ja mahdollisuuteen liikkua. (Autodesk 2016, 1; Unity 2017a, 1.)

Monimutkaisen mallin animointiin on useita eri tapoja. Tapoja voi myös yhdistellä sen mukaan, miten ne sattuvat sopimaan parhaiten animoitavaan malliin tai käyttötarkoitukseen. Taitamalla useita eri tapoja ja kyvyllä yhdistellä niitä on mahdollista saavuttaa paras lopputulos ilman kohtuullisen suurta vaivaa tai ajan käyttöä. (Simmons, Wilhelms & Van Gelder 2002, 145.)

Esimerkkinä eri tavoista on mallin morfaus key framen välissä. Mitä enemmän aikaa ja vaivaa käyttää tässä menetelmässä, sitä luonnollisemmaksi ja yksityiskohtaisemmaksi liikkeen voi saada. Lisäämällä envelopes editoinnin aikana painon jakautuminen ihon eri osa-alueilla, morfaamisen luonnollisuutta voi vieläkin paremmin edistää. Monimutkaisen mallin animoimiseen voidaan käyttää myös ns. hierarkia- tai luurankoanimointia. Hierarkia-animoinnissa käytetään rigid bodyn luustoa, kun liikettä animoidaan. (Simmons, Wilhelms & Van Gelder 2002, 142–145.)

Kolmiulotteisen mallin animoinnissa voi käyttää myös motion capture -tekniikkaa. Motion capture tarkoittaa liikkeenkaappaus menetelmää, jossa muutetaan liikkuvan kohteen liikkeet digitaaliseen muotoon. Motion capture -laitteistoon kuuluu usealla liikesensorilla täytetty puku sekä liikkeen tallentavia kameroita, jotka laskevat puvun eri liikeratoja. Liikkeenkaappausohjelmisto muuntaa nämä liikkeet tietokoneelle. Tätä nauhoitettu liike voidaan suoraan liittää 3D-mallin rigid bodyyn tai luurankorakenteeseen. Tällä tavalla on mahdollista saada mallinnetun hahmon liikkeistä hyvinkin luonnonomaisia ja uskottavia, hyvinkin pienellä vaivalla. (Wikipedia 2017b&c, 1.)

## 2.5 Eksportointi

Eksportointi, tai ”vienti” tarkoittaa datan muuttamista muotoon, joka soveltuu paremmin toiseen ohjelmistoon (Wepopedia 2017, 1).

Eksportointi on tärkeä osa mallinnustyötä, sillä oikein eksportattuna 3D-mallit animaatioineen toimii niin kuin itse mallinnusohjelmassa. Eksportoinnissa on otettava huomioon kohde ohjelmiston vaatimukset, rajoitukset ja suositeltavat tiedostomuodot. Toimivia vaihtoehtoja voi olla useita joista on valittava juuri tarkoitukseen parhaiten toimiva vaihtoehto. Eksportointi vaiheessa on mahdollista rajata tai lisätä eri ominaisuuksia, joita halutaan tuoda toiselle ohjelmistolle. (Autodesk 2017, 1.)

3D-mallien eri eksportointiformaatteja on useita, kuten esimerkiksi .FBX, .OBJ, .DAE tai .MAX, 3D-mallinnusohjelmistoilla on omat ohjelmistokohtaiset formaattinsa ja ne ovat suositeltavia tallennusmuotoja itse 3D-malleille, sillä omissa formaateissaan pysyy kaikki se tieto, mikä on itse 3D-ohjelmistossa tuotettu. Usein nämä tiedosto muodot eivät aina sellaisenaan sovi esimerkiksi peli-

moottoriin, vaan eksportointiasetukset on muutettava. Unity-pelimoottori ymmärtää 3Ds Max:n omaa, .MAX-formaattia, mutta tässä eksport-asetuksessa on kaikki 3D-mallinnus ohjelmiston tieto mukana. Se ei suuren tiedosto kokonsa takia usein sovellu peleihin käytettäväksi sellaisenaan, vaikka eksport-asetuksista on mahdollista karsia eri ominaisuuksia pois 3D-mallista. Pelimoottoreille suositeltu formaatti on .FBX. Tässä formaatissa on vain pelimoottorille kaikki olennainen tieto tallessa, kuten esimerkiksi 3D-malli riggauksineen ja mahdollisine animaatioineen. Myös pelimoottoreita varten formaatti .OBJ on käyttökelpoinen, mutta tämä formaatti ei aina tuo kaikkea tarvittavaa tietoa mukanaan, kuten esimerkiksi animaatioita. (Unity 2017c, 1.)

### 3 VIRTUAALITODELLISUUS JA KEHITYSTYÖKALUT

#### 3.1 Määritelmä

Virtuaalitodellisuus, VR eli "virtual reality" on kolmiulotteinen, tietokoneella generoitu ympäristö, jota voi tutkia ja jonka kanssa voi olla vuorovaikutuksessa. Käyttäjstä tulee osa virtuaalimaailmaa, jossa hän voi manipuloida ympäristöään tai suorittaa erilaisia toimintoja. Laajimmillaan virtuaalitodellisuudeksi voidaan myös mieltää illuusio siitä, että ihminen muualla kuin oikeasti on. (VRS 2016b, 1.)

Virtuaalitodellisuutta ei pidä sotkea toiseen, samankaltaiseen teknologiaan nimeltä "lisätty todellisuus", augmented reality eli AR. Lisätyssä todellisuudessa käyttäjä näkee joko älypuhelimien tai Google Glass -tyylisten datalasien avulla lisättyä informaatiota näkemästään kohteesta. Tämä lisätty informaatio voi olla tekstiä, digitaalista kuvaa tai jopa 3D-malleja. Tässä teknologiassa luodaan illuusiota sinne, missä olet. (Koskenlaakso 2016, 1.)

Virtuaalitodellisuutta voidaan hyödyntää monessa eri asiassa, kuten esimerkiksi peleissä, simulaatioissa ja eri tyyppisissä visualisoinneissa, kuten esimerkiksi datan, arkkitehtuurin tai erilaisten sisätilojen visualisoinneissa (Groningen Yliopisto 2015, 1).

#### 3.2 Virtuaalitodellisuuden historia

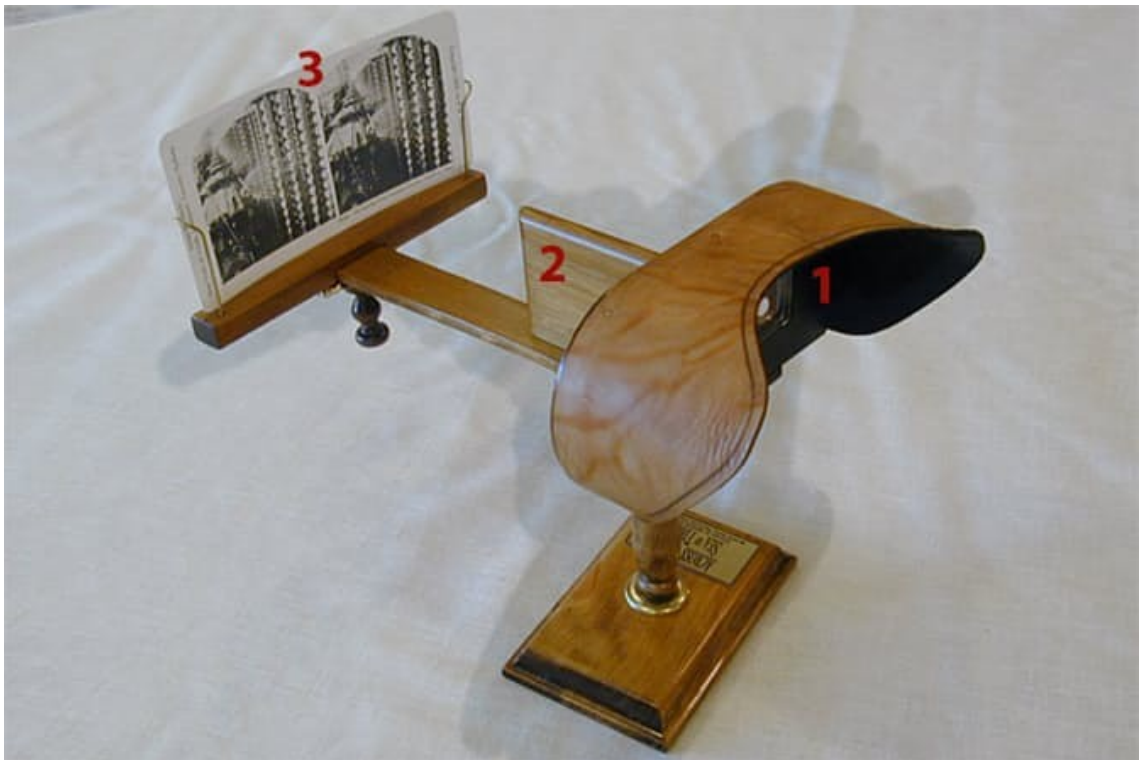
Vaikka virtuaalitodellisuus mielletään hyvin uudeksi asiaksi, virtuaalitodellisuuden verrattavia kuvia ja teknologiaa on pyritty luomaan jo vuosisatoja. Laajimmalla virtuaalitodellisuuden määritelmällä, jossa on luotu illuusio siitä, että olemme toisaalla kuin oikeasti, on kokeiltu ensin esimerkiksi maalaustaiteen avulla. Esimerkkinä tästä on kuvassa 1 Franz Roubaudin panoraamateos "Raevsky Battery during the Battle of Borodino".



Kuva 1. Raevsky Battery during the Battle of Borodino (1812)

Roubaudin maalauksessa on pyritty luomaan illuusio siitä, että teoksen katselija olisi itse sotatantereella mukana. Tämä voidaan mieltää yhdeksi varhaisimmaksi virtuaalitodellisuuden kokeiluksi. (VRS 2016a,1.)

Seuraava edistysaskel virtuaalitodellisuusteknologian kannalta oli ns. stereoskooppi. Stereoskooppi (kuva 2) on laite, jolla pystytään katsomaan stereokuvia. Niissä on vierekkäin kaksi kuvaa, jotka eroavat vain hieman toisistaan, ja pieneltä etäisyydeltä katsottuna kuvat muodostavat kolmiulotteisuuden illuusion. Illuusiosta on nyt ensimmäistä kertaa kolmiulotteisuudelle tuttua syvyysvaikutelmaa mukana, illuusion avulla. Stereoskoopit olivat suosittuja jo 1800-luvun alkupuolella. (VRS 2016a, 1; Historia 2012, 1.)



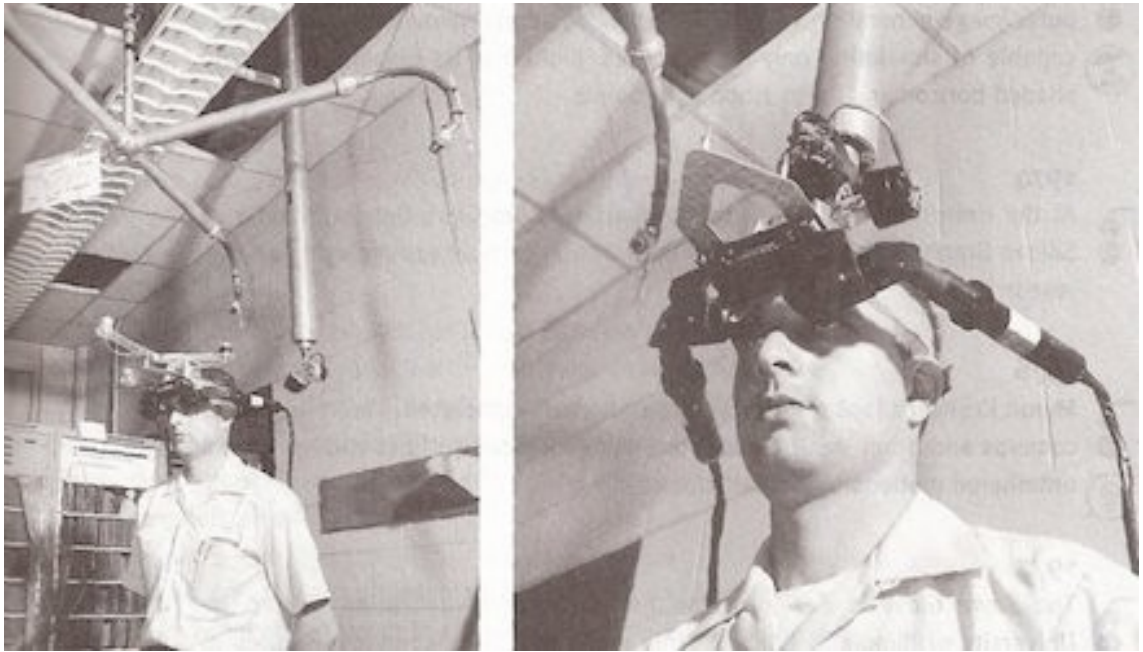
Kuva 2. Stereoskooppi 1850-luvulta (Historia 2012,1.)

Ensimmäisen kerran VR-laseihin viittaavaa teknologiaa esiintyi tieteisfantasias-  
sa 1930-luvulla ilmestyneessä kirjailija Stanley G. Weinbaumin teoksessa "Pyg-  
malion's Spectacles". Kirjassa esitettiin idea laseista, jotka saivat lasien käyttä-  
jät kokemaan fiktionaalisen maailman holografien, tuoksun, maun ja tunnon  
avulla. Weinbaum periaatteessa ennusti jo 1930-luvulla laitteen, joka ilmestyisi  
vasta 90 vuotta myöhemmin. Tämä teki hänestä aikansa todellisen visionäärin.  
(VRS 2016a, 1.)

Vuonna 1960 ilmestyi hieman Weinbaumin laseja muistuttava Morton Heiligin  
keksintö. "Telesphere Mask" oli laite, joka muistuttaa vahvasti jo nykypäivän VR  
laseja. Se oli ensimmäinen päässä pidettävä laitteisto, jolla oli mahdollista kat-  
soa stereoskooppisella tavalla kolmiulotteista elokuvaa. Laseista puuttuivat täy-  
sin stereoäänet, interaktiivisuus sekä pään asennon seuraamiseen vaadittava  
teknologia. Pään asennon seuranta järjestelmä tulikin jo seuraavana vuonna  
"Headsight"-nimisiin laseihin. Headsight-laseja ei ollu kehitetty juuri virtuaalito-  
dellisuutta varten, mutta lasien avulla oli mahdollista nähdä vaarallisia tilanteita  
ja paikkoja armeijaympäristössä ilman, että tarvitsi mennä itse mennä vaaraan.  
Headsight-laitteessa pään liike liikutti samalla kaukana olevaa kameraa, jonka  
avulla käyttäjän oli mahdollista saada luonnollisempi kuva kyseisestä ympäris-  
töstä. Vaikka Headsight oli suuri harppaus nykyajan VR laitteistoon, siitä puuttui  
vielä paljon, kuten esimerkiksi linkki tietokoneen kanssa. (VRS 2016a, 1.)

Vähän myöhemmin, vuonna 1968 tämä puutos korjaantui. Ivan Sutherland ja  
hänen opiskelijansa Bob Sproull kanssa rakensivat ensimmäisen päässä pidet-  
tävän VR-laitteen, joka oli kytköksissä tietokoneeseen eikä kameraan. Laite oli  
kömpelö, todella suuri ja se oli myös kiinnitetty kattoon. He kutsuivat laitettaan  
nimellä "Sword of Damocles", koska katto kiinnikkeiden takia laite näytti hieman  
miekalta, kuten kuvasta 3 on nähtävissä. Käyttäjä oli kiinnitettynä laitteeseen  
eikä voinut juuri liikkua. Tietokoneella tuotettu kuva laseihin oli vielä myös hyvin  
alkukantaisia, "wireframe" eli rautalanka linjoista tuotettuja huoneita ja objekteja.  
(VRS 2016a, 1.)





Kuva 3. Sword of Damocles (VRS 2016a, 1.)

Virtuaalitodellisuus sai nimen ensimmäisen kerran vuonna 1987. Vaikka erilaisia keksintöjä oli jo tähän mennessä tuotettu runsaasti virtuaalitodellisuuden luomiseksi, sai se nimensä vasta paljon myöhemmin. Termiä käytettiin ensimmäisen kerran tieteellisessä julkaisussa Jaron Lanierin toimesta. Hän yhtiöineen myös kehitti monenlaisia eri virtuaalitodellisuuslaseja, joita tavallisen kuluttajan oli mahdollisuus ostaa, tosin todella korkeaan hintaan. Lanierista tehtiin myös myöhemmin vuonna 1992 elokuva, joka toi virtuaalitodellisuus-käsitteen lähemmäksi laajempaa yleisöä. Elokuvassa kerrottiin tarinaa Lanierin virtuaalitodellisuuden kehityksen alusta. (VRS 2016a, 1.)

Vuonna 1993 SEGA kehitti Sega VR -prototyypin, jonka oli tarkoitus tulla myyntiin kuluttajille. Sega VR:ssä oli pään seurantalaitteisto, stereoäännet sekä LCD näyttö. Valitettavasti nämä lasit jäivät prototyyppiä, sillä kehittämisessä tuli todella paljon vaikeuksia. Laseille ehdittiin kehittää jo 4 peliä, vaikka ne eivät koskaan tulleetkaan myyntiin. Seuraava VR kokeilu tuli Nintendolta kuvassa 4 näkyvällä laitteella "Nintendo Virtual Boy" vuonna 1995, joka saatiin myyntiin mutta laitteet olivat niin hankalat käyttää, kalliita ja vain kaksivärisiä että myynti jäi todella pieneksi. Näissä laseissa oli vain musta ja punainen väri. (VRS 2016a, 1)





Kuva 4. Nintendo Virtual Boy (VRS 2016a, 1.)

2000-luvulla VR-teknologia aloitti todella nopean ja massiivisen kehityksen. Tietokone-, älypuhelin-, laitteisto- ja ohjelmistoteknologia kehittyi harppauksin samalla kuin hinnat menivät alas. VR kehittyi harppausmaisesti vuonna 2016, kun Facebook:n ostamaan Oculus Riftiin panostettiin kaksi miljardia dollaria. Samana vuonna virtuaalitodellisuutta lähti kehittämään Valve, Steam:ssa toimivilla HTC Vive -laseilla sekä Sony Playstation VR:llä. Nyt vuosittainen virtuaalitodellisuuden kehitystahti on erittäin vauhdikasta ja vain tulevaisuus näyttää, paljonko teknologia voi kehittyä ja kuinka alas VR-laitteiston hinta mahtaa mennä. (VRS 2016a, 1.)

### 3.3 Ohjelmistot ja laitteet

On olemassa erilaisia VR-teknologian kehittäjiä, kuten HTC Vive, Sony Playstation VR ja Oculus Rift. On myös yhtiöitä, jotka eivät itse niinkään kehitä VR teknologiaa vaan hyödyntävät esimerkiksi älypuhelimia. Muutamia näistä yhtiöistä ovat esimerkiksi Googlen Cardboard ja Samsung Gear VR. Virtuaalitodellisuuden kaltaisen, ”lisätyn todellisuuden” eli AR:n pääkehittäjänä on Microsoft HoloLensillään. (Pänkäläinen 2016, 1; Koskenlaakso 2016, 1.)

Opinnäytteeni keskittyy vain Valve:n HTC Viveen, sillä se oli ensisijainen virtuaalitodellisuuslaite, johon pääsin tekemään visualisointeja Irlannin Ulsterin yli-

opistossa. Vaikka testaukseni ovat tehty pelkästään HTC Vive:n, monet säännöt pätevät myös muihinkin kehittäjiin. Visualisoinnin toimivuus on valmistajasta riippumaton, vaikka esimerkiksi Oculus Rift:llä on ollut alussa ongelmia ns. sumuisten linssien kanssa, joka häiritsi kaikessa visuaalisessa sisällössä.

Koska työni keskittyy juuri pelin visualisoimiseen, on oltava käytössä pelimoottorihjelmisto. Kaksi suurinta pelimoottoria pelintekijöiden keskuudessa ovat Unreal Engine ja Unity. Unity pelimoottorina on saamassa enemmän jalansijaa Unreal:ta, sillä Unreal Engine nostaa rojalteja, kun pelin myynti ylittää 3000 dollaria neljännesvuodessa. Unity taas ei kerää minkäänlaisia rojalteja tehdyistä peleistä yrityslisenssin oston jälkeen. (Unreal Engine 2017, 1; Unity 2017d, 1)

Ohjelmia, joilla itse virtuaaliympäristöjä ja objekteja voidaan luoda, on paljon. Käytännössä mikä tahansa 3D-mallinnus ohjelma tai pelimoottori käy ympäristöjen ja objektien tekemiseen virtuaalitodellisuutta varten. Suosituimpia 3D-mallinnusohjelmia juuri peleihin ovat Autodeskin Maya ja 3Ds Max tai ilmainen Blender-mallinnusohjelma. Pelimoottoreista suosituimpia on edellä mainitut Unity ja Unreal Engines. Opinnäytteeni keskittyy 3Ds Max -3D-mallinnusohjelmaan sekä Unity:n.

Unity pelimoottori on erinomainen vaihtoehto juuri VR pelien tekemiseen, sillä uusimmissa Unityn päivityksissä on mahdollista muuttaa peli VR muotoon erittäin helposti. Unityssä on myös täysi tuki Steam:n ja HTC Vive:n kanssa.

Teksturointeja varten käytin Adoben Photoshop ja Substancen BitMap2Material-ohjelmia. Tekstuurimappien tekemiseen käytin eritoten Photoshop-ohjelmistoa ja esimerkiksi Normal, Specular ja Bump-mappeja tein BitMap2Material-ohjelmistolla. BitMap2Material pystyy tekemään valmiista tekstuurimapista hyvinkin nopeasti omat ja ainutlaatuiset mapit. BitMap2Material-ohjelmiston voi myös lisätä "addonina" eli integroituna Unityyn, jolloin bump-, specular- ja muiden mappien yksityiskohtaisen säätämisen voi tehdä suoraan Unitystä käsin.

Pelin logot ja muut pienet skaalattavat kaksiulotteiset elementit tein ilmaisella vektoripohjaisella ohjelmalla nimeltä Inkscape. Adoben Illustrator on yksiä suosituimpia ja parhaimpia vektoripohjaisia piirros ohjelmia, mutta se ei ollut käytettävissä harjoittelujakseni aikana.

#### 4 VIRTUAALITODELLISUUSPELIN VISUALISOINTI

Ennen kuin lähdin itse visualisointeja tekemään virtuaalitodellisuuspeliin, tutkin mitä aiheesta oli kirjoitettu tai luennoitu. Aloitin tämän tutkimusprosessin aikana, jolloin ainoat HTC Vive -VR-laitteet oli mahdollista saada vain kehityskäyttöön eikä niitä ollut vielä yleisesti myynnissä. Tämä oli yksi syy, miksi en juuri löytänyt kirjoituksia tai julkaisuja valmiiksi testatuista ja hyväksi koetuista menetelmistä virtuaalitodellisuuden visualisoimiseen. Pelejäkin oli vielä tehty niin vähän koko VR-teknologiaan, että ainoa mahdollisuus löytää toimivat menetelmät visualisointiin olisivat omat huomioni sekä ne harvat pelit, joita oli tässä vaiheessa tehty. Tämä teki koko suunnitteluprosessin aloittamisesta hyvin hankalaa, sillä en voinut olla lainkaan varma, mitä kannattaisi virtuaalitodellisuuteen mallinnettaessa ottaa huomioon.

Työharjoittelujaksoni jälkeen opinnäytettä kirjoittaessani VR-pelejä oli jo paljon enemmän ja teknologiakin oli kehittynyt paljon eteenpäin niin Oculus Rift:llä, HTC Vive:llä kuten myös Sony Playstation VR:n osalta.

Koska tutkittua ja testattua tietoa ei ollut saatavilla, turvauduin siihen tietoon, mitä saisin itse VR-pelien pelaamisesta. Tällä tavalla saisin selville, millaisia VR-pelit ovat ja miten itse VR pelaaminen toimii. Ensimmäisiä huomioita olivat peleissä se, että pelit olivat visuaaliselta tyyliltään siloteltuja ja yksinkertaisia. Esimerkiksi HTC Vive:n mukana tuleva The Lab sekä peli Job Simulator olivat hyvin samankaltaisia keskenään. Toinen huomio oli se, että vaikka esimerkiksi Job Simulator:n maailma oli erittäin pelkistetty ja sarjakuvamainen, se ei haitannut virtuaalitodellisuuden kokemista omaksi. VR-lasien kyky liikuttaa näkymää pään eri asennoissa ja ”omien” käsien liikkeen saumaton näkyminen saivat illuusion aikaiseksi siitä, että olisin pelin maailman sisällä. Pelatessa tein myös sen huomion, että jos pelin aikana lähestyy oikean maailman seinää, pelin sisälle ilmestyy hento ruudukko. Ruudukosta sain tiedon siitä, kuinka lähellä oikeaa seinää olin, sekä missä suunnassa se on tulossa vastaan.

Pelien testaamisen jälkeen tutkin, millaisia eri pelialueita ja muita ratkaisuja on tehty pelaajan liikkeeseen liittyen. Pelaajan liikettä seuraavia kameroita HTC Vive:llä oli kaksi, jotka olivat 180 asteen kulmassa toisistaan. Playstation VR:ssä ja Oculus Rift:ssä oli vain yksi liikettä analysoiva kamera. Yhden kame-

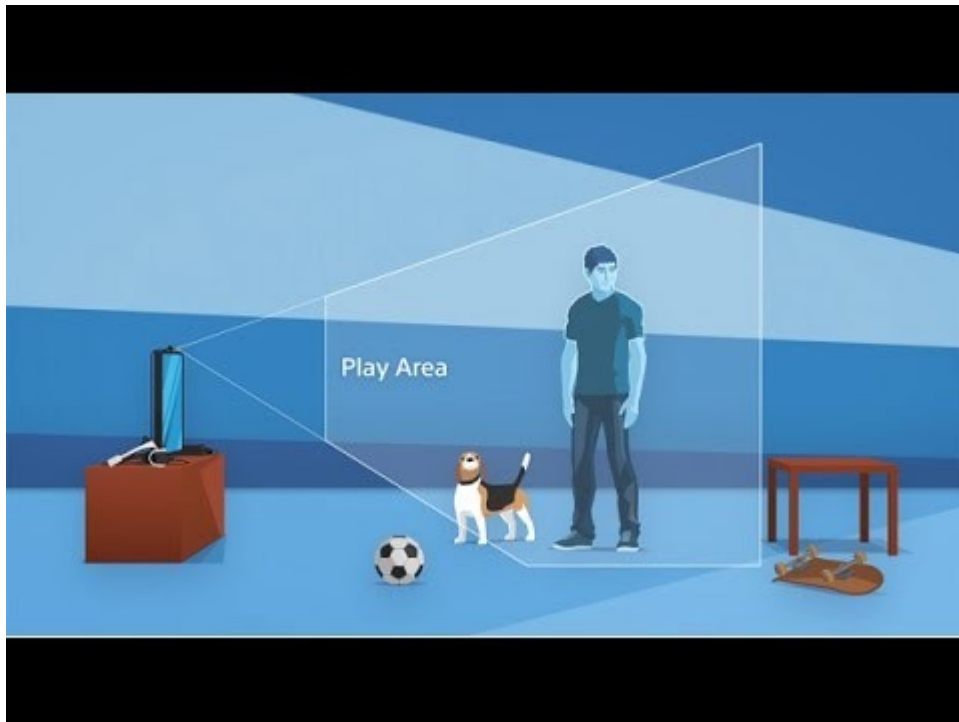
ran järjestelmä kykenee analysoimaan melkein 180 asteessa, kun taas kahdella kameralla on mahdollisuus päästä lähelle 360 asteen analysointia.



Kuva 5. Esimerkki HTC Vive:n pelialueesta (Vive 2017,1.)

Kuvassa 5 on esitetty, kuinka liikettä analysoivat kamerat on aseteltu HTC Vive:llä. Kamerat asetetaan 180 asteen kulmassa toisiinsa nähden, joten pelialueesta tulee neliön mallinen. Esteiden, kuten kehon tai edessä olevan käden takaa kameroiden analysointi voi estyä ja loppua hetkeksi. Tämä aiheuttaa pelissä sen, että käsiä kuvaavat 3D-mallit pysähtyisivät ja palaisivat oletusasentoon niin kauaksi aikaa, kuin kameroiden analysointi on estynyt. Tämä häiritsisi todella paljon virtuaalitodellisuuden illuusiota, joten esimerkiksi Job Simulator -pelissä pelaajan selän taakse ei ole sijoitettu mitään interaktiivista. Suunnittelemalla pelin tällä tavalla voidaan estää kameroiden mahdollisesti aiheuttamat ongelmat.

HTC Vive:n pelaamiseen vaikuttavana tekijänä on laseista selän taakse menevä johtopino. Pelejä pelatessa on mahdollista astua tai jopa kompastua näihin johtoihin. Tähänkin ongelmaan Job Simulator oli keksinyt omanlaisensa ratkaisunsa, jossa pelaajan pelialue oli niin tiukasti rajattu, ettei johdon kanssa tulisi juuri ongelmia. Kun pelaajaa kehoitetaan pysymään hyvinkin pienessä tilassa, ei pelaajalle tule tarvetta pyöriä tai liikkua taaksepäin.



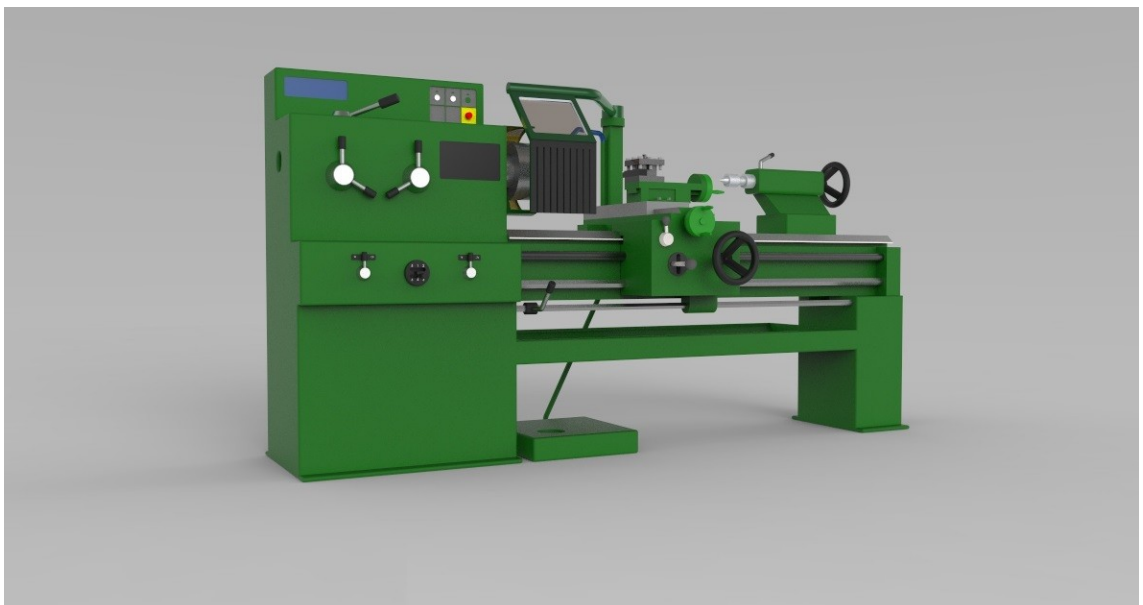
Kuva 6. Esimerkki Playstation VR:n pelialueesta (Tumbokon 2016, 1.)

Playstation VR:llä ja Oculus Rift:llä on yksi liikettä analysoiva kamera pelaajan edessä. Tämä ratkaisu jo itsessään rajoittaa pelialuetta huomattavasti, kun verrataan HTC Vive:n kahden kameran järjestelmään. Kuvassa 6 on hahmoteltu, miltä yhden kameran järjestelmässä pelialue näyttää. Pelialueesta tulee pyramidin muotoinen, niin että kärki osoittaa kameraan ja pohja pelaajaan. Tämä luo tarpeen rajoittaa pelaajan liikettä niin, ettei pelaaja käänny 90 asteen kulmassa kameraan. Jos pelaajalla on molemmat kädet yli 90 asteen kulmassa, liikettä analysoivan kameran analysointi vaikeutuu toisen käden estäessä kameran näkökentän. Tämä tulee ongelmaksi tulevaisuudessa, kun molemmille VR teknologian kehittäjille mahdollisesti tulevat omat käsien liikettä seuraavat ohjaimet. Tällä hetkellä sekä Playstation VR:ää että Oculus Rift:ä pelataan vain tavallisella, liikettä tunnistamattomalla ohjaimella. Molemmissa laitteissa pelataan istuma- tai seisomisasennossa kasvot kameraa kohti. Koska tällaisessa ratkaisussa liike on jo hyvin rajattua, johtoihin kompastumisesta ei juuri tule ongelmia (Tuominen 2016, 1.)

Tästä tutkimuksesta tuli ilmi se, että koska pelini visualisointi tehdään HTC Vive:lle, on minun otettava huomioon kameroiden analysoinnin rajoitteet sekä johtojen tuomat ongelmat. Apuna voin käyttää testattuja pelejäni, joissa nämä ongelmat oli ratkaistu hyvinkin johdonmukaisesti.

Seuraava tehtävä oli tutkia aihetta, josta peli tehtäisiin. Visualisoitavan pelini aiheeksi kehkeytyi sorvin käyttöön tutustuttava opetuspele. Tätä tutkimusta varten pääsin tutustumaan oikeaan sorviin, siihen liittyviin työkaluihin sekä tiloihin. Otin paljon valokuvia sekä sorvista, työkaluista että tiloista, joita voisin jälkeempään käyttää apuna visualisoinnissa ja mallintamisessa. Tästä tutkimuksesta tuli hyvin ilmi, mitä kaikenlaisia työkaluja liittyy sorvin käyttöön sekä sorvin toimintaperiaatteet. Sorvin ammattikäyttäjän haastattelusta opin myös paljon siitä, mitä on otettava huomioon, jotta sorvia voi käyttää turvallisesti ja mitkä ovat yleisimmät virheet sorvin käytössä. En ollut varma, mikä kaikki tieto olisi olennaista omaan visualisointiin, mutta perehdyin sorvin käyttöön liittyviin asioihin sen verran perinpohjaisesti, että osasin jo nimetä kaikki sorvin osat englanniksi. Oivallus, jonka kokosin kaikesta tästä tiedosta, oli se, että VR-peli olisi erinomainen tapa opettaa tämän koneen käyttöä. Virtuaalitodellisuus mahdollistaa tämän erittäin vahvaan ja vaaralliseen koneeseen tutustumisen ilman hengenvaaraa käyttäjälle.

Koska olin peliprojektimme ainoa visuaalisen sisällön tuottaja, osa sisällöstä hankittiin kolmansielta osapuolilta. Niin rajatussa ajassa, kun työharjoitteluni oli, ei yhden ihmisen ollut mahdollista tuottaa tarpeeksi laadukasta visuaalista materiaalia peliin. Ensimmäiseksi hankimme peliin sorvin 3D-mallin Unityn asset-kaupasta, joka on nähtävissä kuvassa 7. Vaikka hankkisimme asetteja kolmansielta osapuolilta peliin, olisi minun tehtäväni silti tuottaa kaikkiin objekteihin yhtenäinen visuaalinen tyyli tekstuureilla.



Kuva 7. Assettina hankittu sorvin 3D-malli

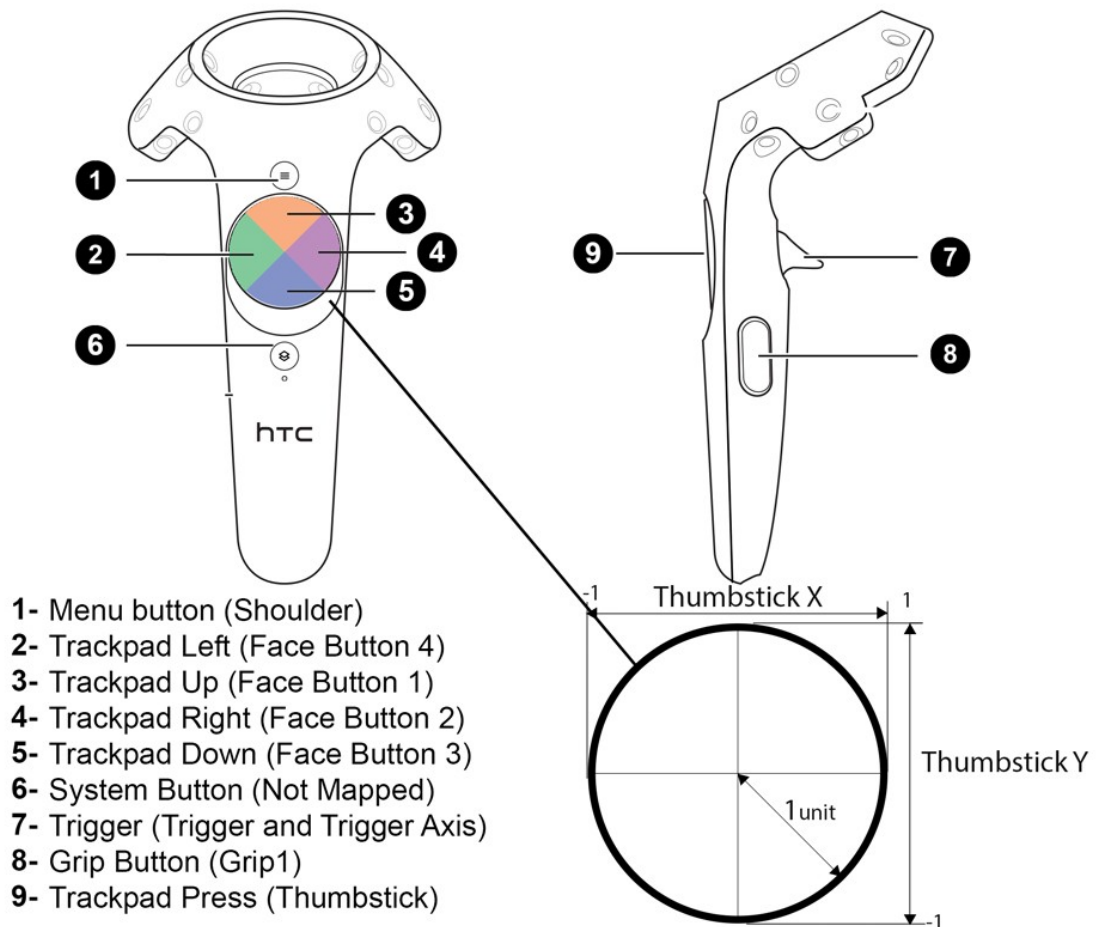
Hankkimamme sorvin 3D-malli koostui monesta eri palasesta. Kaikki erilliset osat olivat omina tiedostoinaan, joten ensimmäiseksi kokosin sorvin kaikkine osineen yhteen 3Ds Max -näkymään. Tämän jälkeen eksporttasin sen sellaisenaan omaan Unity-näkymään ja lähdin suoraan testaamaan, miltä se näyttäisi HTC Vive:n VR-lasit päässä. Sain heti huomata sen, että kootun sorvin 3D-malli oli niin raskas, että näkymä hidasteli pahasti VR:ssä. Tämä aiheutti paljon huonovointisuutta ja pystyssä pysyminen oli vaikeaa, kun horisontti näkyi viiveellä. Jo tämä ensimmäinen testi teki selväksi sen, että pelin on sujuttava saumattomasti, jotta pelaaja ei menettäisi tasapainoaan, puhumattakaan pahoinvoinnista. Myös se oli selvää, että 3D-malleille on tehtävä paljon muutoksia polygonimäärään, jotta näitä ikäviä seurauksia ei tulisi vastaisuudessa.

Tutkiessani ongelmaa syvemmin sain selville, että HTC Vive -pelin on toimittava vähintään 75 kuvaa sekunnissa. VR-laseissa on kaksi näyttöä, joiden kuvat poikkeavat vain vähän toisistaan, kuten stereoskoopissa, mutta liikkuvana kuvana. Kaksi näyttöä tarkoittaa myös sitä, että se vie kaksi kertaa enemmän resursseja kuin esimerkiksi tavalliselle, yhdelle näytölle visuaalista sisältöä tehtäessä. Jos VR-pelissä tulee hetkiä, joissa ei saavuteta vähintään 75 kuvaa sekunnissa, pelaajan näkymään tulee viive, joka saa kokemani kaatumis efektin aikaiseksi. Tämän jälkeen oli päivänselvää, että peli on tehtävä niin polygoneiltaan taloudellisesti kuin mahdollista, jotta peli sujuisi moitteettomasti.

Kun tutkin sorvin 3D-mallia tarkemmin 3Ds Max:ssa, huomasin kuinka raskas se todella oli. Täysin kootussa mallissa oli yli kaksi miljoonaa polygonia. Tavallisesti pelissä, joka on tehty yhdelle tietokoneruudulle, yksityiskohtaisen päähahmon polygoni lukema voi yltää n. 20 000:n. Tähän verraten oli sorvini täysin sopimaton sellaisenaan VR-peliin. Lähdin työstämään "lowpoly" eli vähäpolygoniversioita jokaisen sorvin osasta. Käytin apunani 3Ds Max:n ProOptimizer-toimintoa, joka mahdollisti valmiin mallin polygoni lukeman vähentämisen. Vähensin polygonien määrää jokaisessa palassa niin paljon, kuin pystyin ilman, että 3D-mallin muoto muuttuisi. Kun olin tehnyt jokaisesta sorvin osasta lowpoly-versiot, kasasin mallin yhdelle näkymälle. Sorvin kokonaispolygonimäärä putosi parista miljoonasta, n.100 000:n. Malli oli vieläkin raskas verrattuna antamaani esimerkkiin, mutta en saanut polygoni määrää enää alemmaksi, ilman että osien muoto muuttuisi. Eksporttasin uuden kootun sorvin mallin Unity:n ja kokei-



lin, miten se toimisi VR-laseilla. Sain ilokseni huomata, että näkymä ei enää hidastellut ja ongelmat huonovointisuuden ja tasapainon kanssa olivat kadonneet.



Kuva 8. HTC Vive:n käsiohjain ja niiden toiminnot (Proteus 2015, 1.)

Seuraavaksi lähdin tutkimaan sitä, miten HTC Vive:n ohjaimet toimivat. HTC Vive:llä käytetään ohjaimina kapulaa kummallekin kädelle, joissa on 5 nappia. Kunkin napin toiminnot on mahdollista liittää haluttuun ominaisuuteen koodin avulla. Kuvassa 8 on nähtävissä nappien perusominaisuudet ja ohjaimen muoto. Olin oppinut joitakin asioita pelikapuloiden toiminnoista, kun kokeilin eri VR-pelejä. Kaikissa peleissä, joita kokeilin, oli liikkuvuutta helpottava teleporttaustoiminto. Tämä toiminto oli suuressa napissa kapulan päällä, kuvan 8. numeroissa 2-5. Painamalla kerran napista, tuli pelin sisällä tähtäin, jolla voitiin valita etäisyys ja suunta. Toisella napin painalluksella pelaaja teleporttasi tähdätyyn kohteeseen. Tämä toiminto mahdollisti suuren alueen tutkimisen, ilman että pelialueen tarvitsisi olla kovin suuri pitkän kävelymatkan takia. Kokeilluissa peleissä kaikki interaktiivinen toiminto oli kohdistettu 7. liipaisimeen. Vetämällä liipaisi-

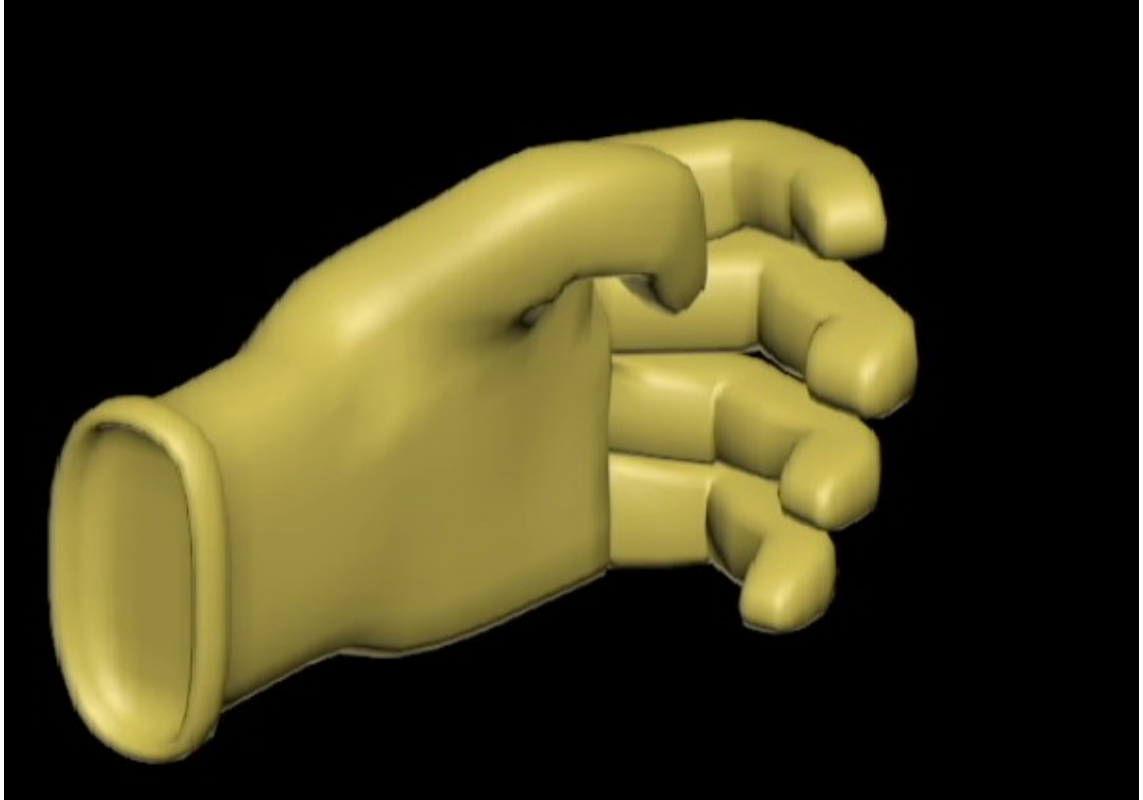


mesta oli mahdollista tarttua esineisiin ja interaktiivisesti kokeilla ympäristöä. Näiden kokemusten ansiosta oli helppo aloittaa kapuloiden ominaisuuksien valikointi, joissa liipaisimesta vedettäessä tapahtuisi kaikki interaktiivinen ja suuresta napista teleporttaus. Jo nämä kaksi toimintoa mahdollisti kaiken, mitä pelisämme tällä hetkellä tarvitsimme. Kuvan 8. muita nappeja emme laittaneet vielä tässä vaiheessa käyttöön.

Kun käsiohjainten toimintaperiaatteet oli selvillä ja nappeihin oli valittu toiminnot, lähdin tutkimaan, miten tekisin omat 3D-mallit käsiohjainten tilalle. Kun Unity:ssä testaa tyhjää näkymää VR-laseilla, näkyvät HTC Vive:n omat ohjaimet ”käsinä” pelin sisällä. Tehtäväni oli korvata nämä ohjaimet ihmisen käsillä. Pyrin tekemään kädet niin, että ne sopisivat mahdollisimman monelle eri tyyppiselle ihmiselle, jotta samalla peli sopisi suuremmalle yleisölle. Mitä olin oppinut sorviin liittyvistä tutkimuksista oli se, että käsineitä ei saa missään nimessä käyttää. Käsineet voivat takertua sorvin pyöriviin osiin ja vahingot voivat olla mittavia. Tämä oli sikäli harmillinen rajoitus, sillä hansikkaat olisi ollut helppo tehdä niin, että se sopisi kaiken tyyppisille pelaajille, kun ihonvärin ja käden tyypin voisi paremmin piilottaa.

Lähdin suunnittelemaan käsiä, jotka olisivat sen verran geneeriset, että ne voisivat olla sekä miehen että naisen kädet. Olin saanut ohjaajaltani ohjeistuksen, jonka mukaan peli saisi olla tyyliltään semi-realistinen ja hauska. Näillä tiedoilla suunnittelin käsiä, jotka olisivat hieman sarjakuvamaiset tuomaan hieman hauskuutta sekä piilottamaan käsien sukupuolieroja. Tein ensin 3Ds Max:ssa laatikko mallinnus -menetelmällä yhden käden valmiiksi, jota lähdin sitten muovamaan niin, että se näyttäisi hieman sekä miehen että naisen kädeltä. Tässä käytin apuna Smooth-modifieriä. Tein kämmenestä hieman laatikkomaisen ja sormista taas sirohkot. Eksporttasin käden mallin .FBX tiedostomuotona Unity:n ja asetin käden ohjaimen tilalle. Kun testasin kättä VR-lasien kanssa, sain huomata kuinka pieleen käden koko oli mennyt. Käsi oli suhteessa aivan liian iso ja pitkä. 3Ds Max:ssa käsi näytti hyvin sopusuhtaiselta, mutta tein muutokset ja eksporttasin uuden mallin ja testasin VR-laseilla. Tällä kertaa koko oli lähellä normaalia ja käsi näytti muutenkin sopusuhtaisemmalta. Onnistuneen testin jälkeen lähdin riggaamaan. Tein ensiksi käden luuston mallin sisälle. Envelopes ja Weight mappauksen jälkeen tarkistin vielä VR:ssä, muuttuiko mikään näiden työvaiheiden jälkeen. Käsi näytti samanlaiselta, joten lähdin luomaan animaatio-

ta. Animaation tarkoitus olisi näyttää, että pelaaja on vetänyt liipaisemista. Koska interaktiiviset toiminnot olisivat liipaisimessa, tapahtuisi tarttumis-animaatio. Tein tästä animaatiosta hyvin yksinkertaisen, jossa jokainen sormi koukistuisi hieman kämmentä kohti kuten kuvan 9 lopputuloksessa.



Kuva 9. Käden 3D-mallin tarttuminen

Testasin animoidun käden Unity:n puolella. Käsi näytti hyvältä, mutta animaatio oli vielä hieman tökerö. Animaatio oli liian hidas, ja pari sormea ei liikkunut luonnollisesti. Korjasin tämän viimeksi mainitun ongelma sillä, että vedin etusormea eteenpäin ja keskisormea taaksepäin, luoden luonnollisemman näköisen koukistuksen. Eksporttasin ja testasin VR:ssä, mutta animaatio ei vielääkään osunut täysin kohdilleen joten hioin vielä toistamiseen animaatiota, kunnes olin tyytyväinen lopputulokseen. Tämän jälkeen tein Mirror-komennolla oikean puolen käden ja siihen nimesin luut oikeaan käteen sopiviksi. Kun testasin molemmat kädet, ne toimivat ja näyttivät suhteellisen luonnollisilta. Kädet jäisivät ilman tekstuureita, sillä vielä en ollut päättänyt pelin lopullisesta visuaalisesta suunnasta. Värin ne saivat Unity:n oletus-valkoisesta alustavasti. Tein kuitenkin unwrap UVW:n molempiin käsiin valmiiksi.

Nyt kun pelissä oli sekä sorvi että kädet, tarvitsimme jotain, johon tarttua tai heittää. Koodi puolen tekijät tarvitsivat objekteja, joita voisi heitellä ja testata muita toimintoja. Koska tiesin, että tulisimme tarvitsemaan useita työkaluja tulevaisuudessa, päätin tehdä koodareille työkaluja kokeiltavaksi. Tein yksinkertaiset, hyvinkin lowpoly-mallit ja teksturoin ne alustavasti kuvan 10 mukaisesti.



Kuva 10. Ote muutamista 3D-mallinnetuista työkaluista

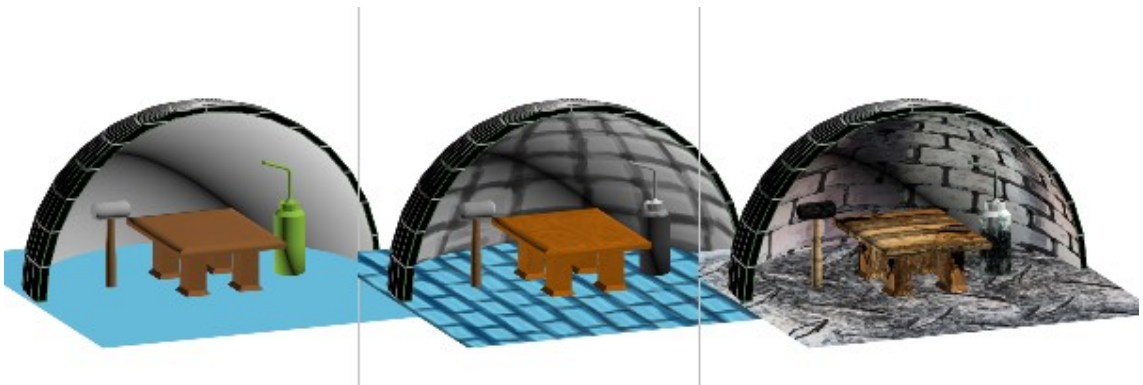
Mallinsin useita eri työkaluja, joita olin nähnyt oikean sorvin tiloissa sekä saamistani ohjemateriaaleista. Vaikka en ollut vielä päättänyt pelin visuaalista suuntausta, tein unwrap UVW:t ja teksturoinnit kaikkiin työkaluihin. Tein tekstuurit tyyllillä, jolla niitä olisi helppo muokata sitä mukaa kun tyyli selkeytyisi. Kaikki yksittäiset elementit olivat omilla tasoillaan eli "layer:llä" Photoshop:ssa. Testasin paria eri työkalua VR:ssä, mutta en pystynyt juuri tekemään johtopäätöksiä, sillä tekstuurit tulisivat muuttumaan tulevaisuudessa. Ei ollut myös mitään taustaa, mihin työkaluja voisi verrata, muuta kuin itse vihreä sorvi. Työkalut olivat kuitenkin hauskan näköisiä VR-lasien kanssa.

Nyt pelissä oli hieman enemmän sisältöä, joten oli aika päättää se, millä visuaalisella tyyllillä jatkaisin koko pelin loppuun. Ohjaajani oli päättänyt, että tyyli saisi jatkua samana, eli suhteellisen realistisena, mutta samalla hauskana. Ohjaaja ohjeisti tekemään The Lab:n kaltaisen visuaalisen tyylin. Koska lopullisesta visuaalisesta tyylistä tehtäisiin äänestys työryhmäni kesken, olisi tehtävänäni luoda monta eri vaihtoehtoa, joista valita. Haasteena tulisi keksiä tapa, millä näyttää eri suunnitelmani ymmärrettävästi muille. Päädyin ratkaisuun, jossa käyttäisin jo pelistä löytyviä objekteja ja laittaisin ne yksinkertaiseen tyhjään näkymään 3Ds Max:ssa. Valitsisin pari erilaista työkalua ja näyttäisin, miten ne sopisivat ympäristöönsä. Ympäristön tekisin hyvinkin yksinkertaisesti, jossa olisi lattia ja kupu joka toimisi sekä kattona että seininä.



Kuva 11. Perusnäkö, ympäristöineen ja työkaluineen

Kun perusnäkö oli valmis ja havainnollistavat tekstuurit oli haettu, kuten kuvassa 11 näkyy, alkoi eri tyylien luomisprosessi. Tein hyvin erilaisilla tavoilla, erilaisia tyyliä, kuitenkin pitäen mielessä sen, että pelin olisi oltava The Labin tapainen sekä semi-realistinen ja hauska. Osan tyyleistä piirsin itse, osassa käytin hyväkseni Photoshop:n artistic-tyylejä. Kokeilin kaikenlaisia eri variaatioita ja loppujen lopuksi loin 17. erilaista näkymää. Näistä 17. eri tyylistä valitsin 10 parasta, jotka esittelin yksitellen ryhmäni jäsenille. Kuvassa 12 on ote kolmesta eri tyylistä.



Kuva 12. Ote kolmesta erilaisesta näköstä tyylistä



Kuva 13. Lopulliseksi valittu visuaalinen tyyli

Vaikka tyylejä oli paljon hyvinkin erilaisia, ryhmäni päätyi hyvinkin yksiselitteisesti kuvan 13 näkymään. Mielestäni tämä valituksi tullut tyyli oli hyvin lähellä sitä, mitä ohjaaja oli minulta vaatinut. Suurien väripintojen muodostama hieman sarjakuvamainen, mutta realistinen tyyli myös muistutti The Lab:n visuaalista tyyliä. Tyyliässä oli myös hyvin latteat värit, niin kuin myös muissa testatuissa peleissäkin. Nyt kun visuaalinen tyyli oli valittu, lähtisin seuraavaksi muokkaamaan kaikki tekemäni tekstuurit näyttämään samankaltaisilta. Vielä en tehnyt muita kuin tekstuuri mappoja, joten esimerkiksi bumb-, normal- ja specular-mapit odottaisivat sitä, että pelissä olisi enemmän materiaalia. Kun olin saanut usean objektin teksturoitua valitulla tyyllillä, eksporttasin ne Unity:n ja kokeilin, miltä ne näyttäisivät VR-laseilla. Sain huomata, että tyyli näyttää aika hyvältä virtuaalitoellisuudessa, mutta esimerkiksi kuvan 13. pöydän pienet viivat vilkkuivat ja värisivät hyvin häiritsevästi näkökentässä. Tämä sai pahoinvointia aikaiseksi, kun pienet linjat vilkkuivat sitä tahtia, kun muutin pääni asentoa. Tämä käytännössä tarkoittaa sitä, että ne vilkkuisivat kokoajan pelissä. Jos samanlaisia vilkkuvia

kohteita olisi enemmänkin pelissä, se varmasti tuottaisi huonovointisuutta ja pahimmassa tapauksessa jopa epilepsiakohtauksen. Mutta kaikki muu, latteaa ja suuri kuvioinen pinta toimi ilman ongelmia. Kokeilin muuttaa pöydän tekstuuria suuremmiksi väripinnoiksi ja näin värinä ja vilkkuminen hävisi VR:ssä. Seuraavissa teksturoinneissa on pidettävä huolta siitä, ettei kapeita raitoja pääse esiintymään. Sain huomata sen, että ongelmat tulevat vastaan vain kokeilemalla, joten pyrin katsomaan jokaisen elementin jonka teen, aina myös VR-laseilla. Vaikka kaikki näyttääkin hyvältä Unity:ssä ja 3Ds Max:ssa, siitä ei ole mitään varmuutta, miten asiat kokee itse pelin sisällä ollessa.

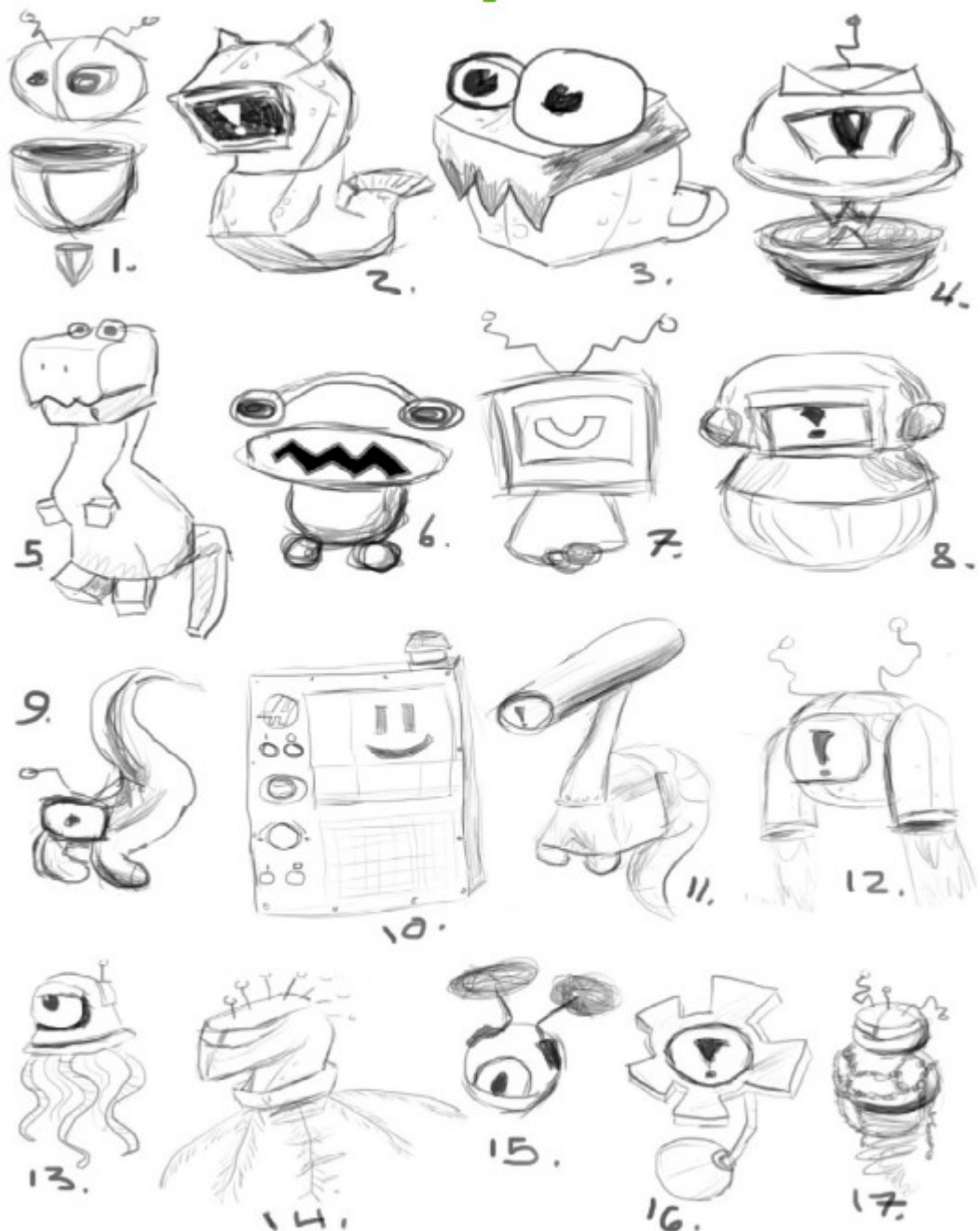
Kun testasin eri objekteja tekstuurineen VR:ssä, sain huomata myös sen, kuinka vaikea oli saada objektien koko oikeaksi. Unity-näkymässä kaikki ovat suhteessa toisiinsa hyvässä tasapainossa, mutta kun peliä testaa VR-laseilla, objekteissa ei ollut enää samaa tasapainoa. Etäisyydet sekä koot olivat hyvin eri suhteessa toisiinsa. Oli vaikea arvioida vain Unity-näkymän perusteella, miltä objektit näyttäisivät VR:ssä. Vaikka pelissä ei vielä ollut montaa erilaista objektia, se oli silti erittäin haastavaa saada ne tasapainoisiksi toisiinsa nähden.

Kun tekstuurien muokkaukset ja hiomiset oltiin saatu valmiiksi, tehtäväkseni tuli suunnitella hahmo, joka opastaisi pelaajaa sorvin käytössä ja muissa toiminnoissa. Ryhmäni jäsenet ehdottivat täysin animoitua, ihmisen 3D-mallia, mutta ainoana visuaalisena sisällön tekijänä minun oli pakko kieltää suunnitelma tällaisenaan. Ihmisen 3D-malli on hyvin monimutkainen ja suurin ongelma tulisi animoinnissa. Pelkästään hahmon animoimiseen menisi enemmän aikaa, kuin meillä olisi koko pelin tekemiseen. Ehdotin jotain yksinkertaisempaa hahmoa, jossa ei olisi montaa animoitavaa osaa, kuten esimerkiksi robottia. Robotissa olisi myös se etu, että sen mahdollisen puheen voi hoitaa paljon helpommin ilman, että ihmisen ääntä tarvitsisi yrittää äänittää luonnollisesti. Robotin äänen voi tehdä ohjelmalla, joka lukee synteettisellä äänellä tekstiä. Ohjaaja piti ajatuksesta ja yhdessä päätimme, että robotti voisi leijua ympäristössä ja se olisi luonteeltaan vähän sekä hauska että outo. Leijuvassa robotissa ei tarvitsisi animoida jalkojen liikkeitä.

Koska robotista tulisi taas äänestys ryhmäni kesken, pyrin visualisoimaan ideoitani niin, että muutkin pystyisivät osallistumaan valitsemiseen. Päätin tehdä sarjan hyvin erilaisia luonnoksia robotista, jossa olisi halutut ominaispiirteet, kuten



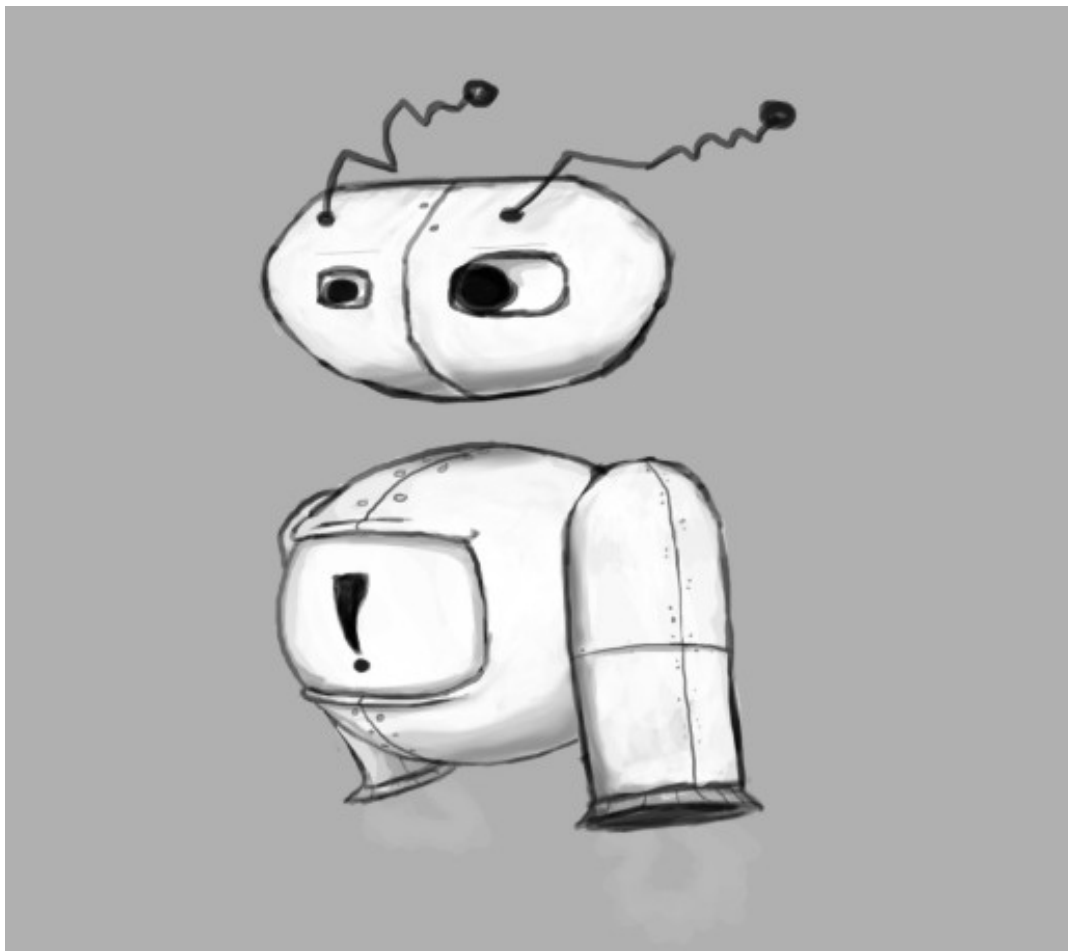
leijuminen, hauskuus ja outous. Lähdin tekemään luonnosteluprosessia Photoshop-ohjelmistossa niin, että piirsin nopeasti ja tyyllittelemättä niin monta ideaa, kuin sain yhdelle paperille mahtumaan. Luonnoksista tuli hyvin karkeita, mutta monipuolisia. Kun luonnokset olivat valmiina, numeroin ne, jotta olisi helpompi keskustella ryhmäni kanssa, mikä luonnoksista miellyttäisi eniten. Kuvassa 14 on nähtävillä kaikkien eri robotti-luonnosten versiot.



Kuva 14. Robottien luonnokset

Robotti-luonnoksia kertyi yhteensä 17 kappaletta. Nopeassa luonnosteluprosessissa piirsin kaikki ideat sellaisenaan ja nopeasti paperille. Lähdin eri muodois-

ta, sorviin kuuluvista työkaluista ja osista, sekä aivan satunnaisista aiheista. Pidin myös mielessäni halutut ominaisuudet. Kun ryhmäni kanssa keskustelimme luonnoksista, päädyimme hyvin mielenkiintoiseen ratkaisuun. Päätimme ottaa nr.1 robotin päänmuodon, ja yhdistämään sen nr. 12. Tämä oli erittäin hyvä ratkaisu, sillä nyt pää olisi hassun ja oudon näköinen sekä kehossa olisivat raketit, jotka selittäisivät robotin kyvyn leijua. Kun robotin visualinen ilme oli päätetty, lähdin pikaisista luonnoksista tekemään kunnon konseptitaiteen. Loin konseptikuvaa robotista sillä silmällä, että se olisi myös apuna itse mallinnusprosessissa. Piirtämällä selkeästi hahmon muodot ja yksityiskohdat mallintamisesta tulisi paljon helpompaa.

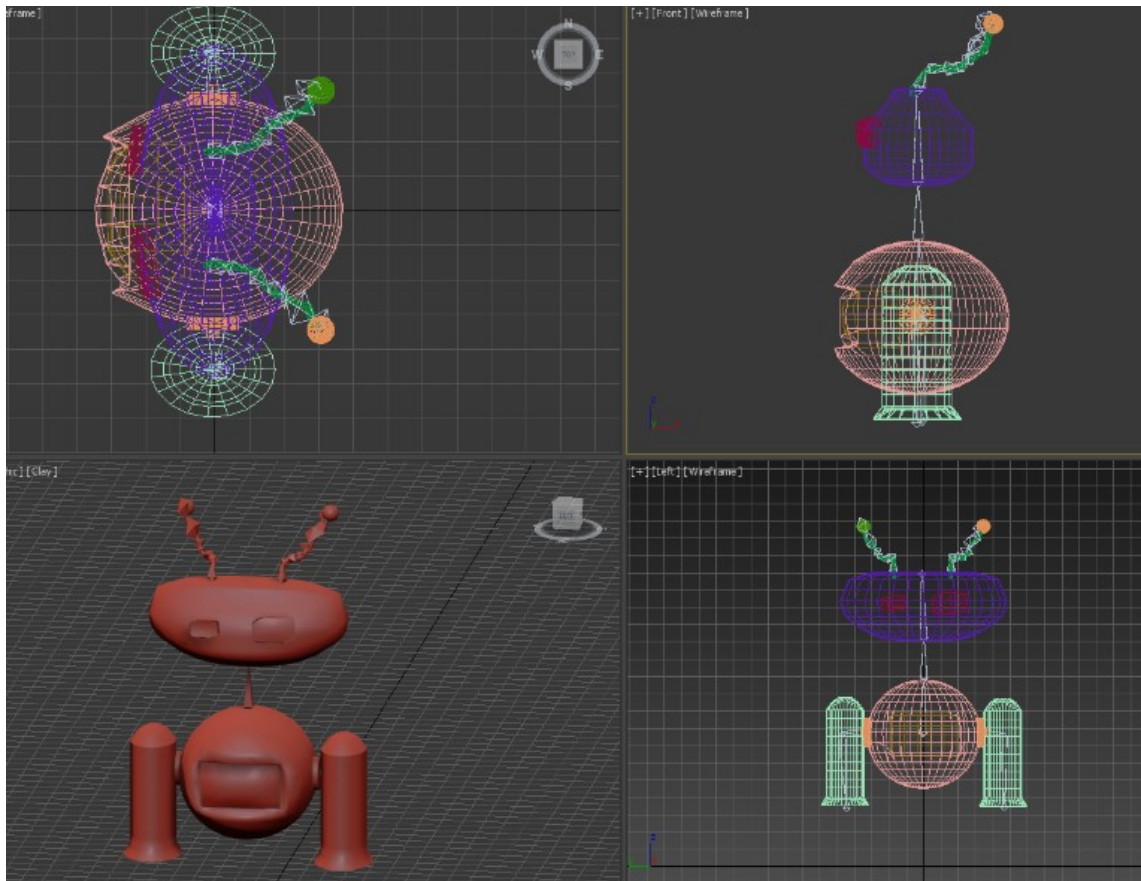


Kuva 15. Robotin valmis konsepti

Kuvan 15 robotin konseptitaiteen luomisessa käytin Photoshop-ohjelmaa. Tein teoksesta selkeä linjaisen ja käytin varjostuksia, jotta robotin muoto tulisi esiin selkeämmin. Pidin värityksen harmaasävyisenä, jotta mahdollisten värien kokeilu jälkeinpäin olisi helppoa tehdä pelkällä Color layer -ominaisuudella. Tein taustasta tummemman, jotta vaalea hahmo näkyisi selkeästi.



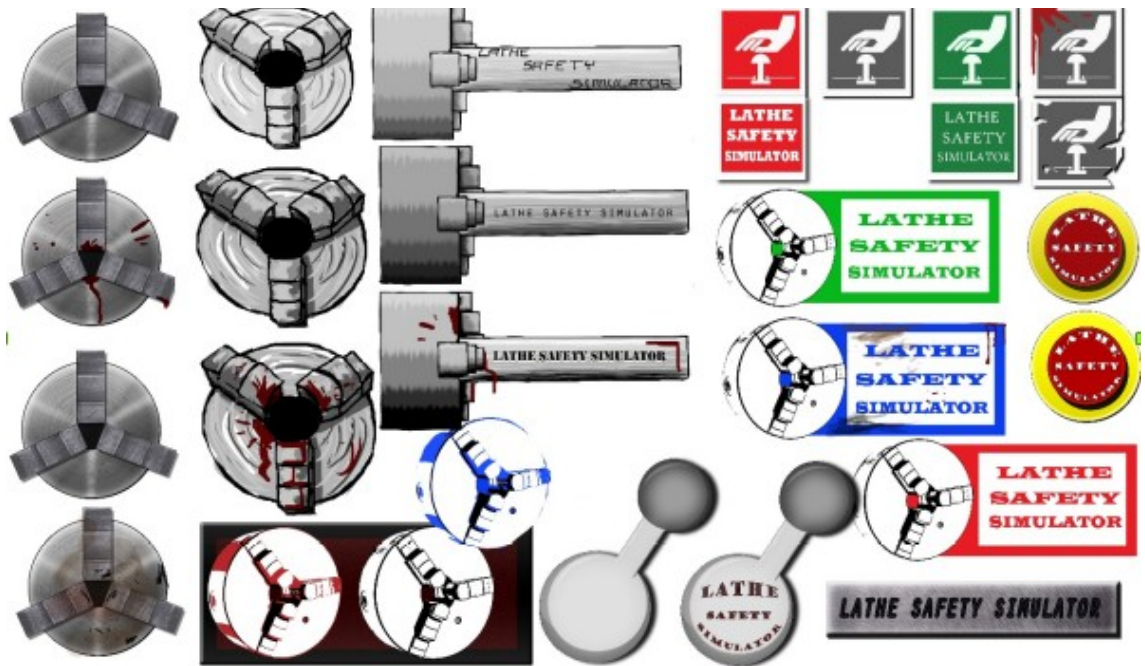
Konseptin valmistuttua aloitin 3D-mallinnusprosessin 3Ds Max:ssa. Lähdin yksinkertaisista muodoista ja robotin piirteet valmistuivat nopeasti. 3D-mallin tekeminen meni hyvin, mutta seuraavassa vaiheessa tuli ongelmia. Kun lisäsin luut ja suoritin muut rigging-prosessiin kuuluvat työvaiheet, ei robotti toiminut halutulla tavalla. Hahmon antennit litistyi luiden mallisiksi. Poistin luut ja lisäsin ne uudestaan, mutta sama ongelma tapahtui uudestaan. Minulla ei ollut tarpeeksi aikaa siihen, että selvittäisin mikä tämän aiheutti, joten tein kompromissin, missä antennit eivät liikkuisi lainkaan. Robotin liikkuviksi osiksi täten jäisi raketit ja pää. Minun ei tarvitsisi animoida robottia itse, sillä sen tekisivät muut Unity:n avulla. Onnekseni kuitenkin testasin robotin liikkuvuuden kaikissa ääriasennoissa ennen kuin eksporttasin sen Unity:n. Pyöritellessä eri luita huomasin, kuinka huolimattomasti olin tehnyt envelopes ja weight mappauksen. Esimerkiksi kun pyöritin robotin raketteja ääriasentoon, huomasin, kuinka robotin sisälle jääneet polygonit menivät liikkeen mukana ja tulivat ihon läpi näkyviin. Samaa tapahtui, kun liikutin päätä ääriasennosta toiseen. Onnekseni olin jättänyt antennit liikkumattomiksi, joten minulla oli vielä aikaa korjata mallin liikkeet. Jos tämä virhe olisi jäänyt valmiiseen peliin, olisi se ollut erittäin epäammattimainen ja visuaalisesti karu.



Kuva 16. Robotin 3D-malli luineen 3Ds Max -näkyvässä

Virheiden korjaamisen jälkeen tein mallille valmiiksi myös unwrap UVW -kartan. Kuvassa 16 näkyy malli luineen ja valmiilla unwrappauksella. Robotin 3D-malli eksportattiin Unity:n ilman tekstuureita, sillä sitä tarvittiin nopeasti käyttöön. Koska olin tehnyt unwrappauksen valmiiksi, olisi tekstuurin lisääminen jälkeensä helppoa. Tämän kiireellisen mallin jälkeen minulle jäi aikaa luoda lisää visuaalista sisältöä peliin. Tein useaan sorvin osaan tekstuureita ja loin uusia työkalujen 3D-malleja. Sain myös yhtenäistettyä valmiita tekstuureita keskenään paremmin sopiviksi.

Sain koko sorvin teksturoinnit valmiiksi ja seuraava tehtävänanto oli luoda pelille logo. Vaikka logo ei itsessään kuulu mallintamiseen millään tavalla, kuuluu se kuitenkin pelin tekemiseen ja sen visualisoimiseen. Logo on tärkeä osa peliä tunnistettavuuden ja brändin kannalta. Logolla voidaan myös yhtenäistää visuaalista sisältöä keskenään. Logon tekemisen ohjeistus oli aivan sama, kuin itse pelin visuaalisen tyylin, eli semi-realistinen, hauska ja opetuksellinen. Sen olisi tietysti sovittava peliin saumattomasti. Ohjaaja ilmoitti pelin lopulliseksi nimeksi ”Lathe Safety Simulator” eli sorvin turvallisuussimulaattori. Sain luomisprosessille muuten täysin vapaat kädet.



Kuva 17. Pelin logo luonnoksia

Logojen suunnittelu on ollut itselleni aina erittäin haastavaa, joten ideoiminenkin oli aluksi hyvin hidasta. Lähdin Photoshop:ssa tekemään loogisesti erilaisia ehdotuksia, sorvin eri osista ja muusta sorviin liittyvästä. Tein ehdotuksia myös sorvin ympäristöön kuuluvista elementeistä, kuten esimerkiksi varoituskylteistä. Koska ideoinnissani oli paljon ongelmia, tein samoista luonnoksista useita versioita. En ollut tyytyväinen luonnoksiini, mutta valmistelin niistä puolet ja näytin ryhmälleni kuvan 17 mukaisesti. Olin unohtanut numeroida eri vaihtoehdot, joten tietystä logosta puhuminen osoittautui haastavaksi. Pääsimme kuitenkin yhteisymmärrykseen ja valitsimme logon, jossa on sorvin pyörivä osa ohjetaulun vieressä sinisellä pohjalla. Ryhmäni jäsenet halusivat siihen vielä lisää likaa ja verta. Näillä ohjeilla lähdin luomaan logoa Inkscape-ohjelmistolla, sillä harjoittelupaikassani ei ollut muuta kuin Photoshop ilman Illustratoria. Logojen tekemiseen on parasta käyttää vektoripohjaista piirustusohjelmaa, jotta logoa voi skaalata loputtomiin ilman, että laatu kärsii.



Kuva 18. Logon ensimmäinen versio

Tein ensimmäisen version logosta ja sain huomata, että lika ei sopinut siihen lainkaan, joten jätin ne pois. Logossa oli jo paljon erilaisia elementtejä. Fontin valitsin muistuttamaan muita ohjetekstejä, joita varoituskylteissä oikeasti on. Näytin kuvan 18 versiota ohjaajalleni, ja hän halusi minun muuttavan sitä, määrittelemättä miten. Kun katsoin uudestaan lopputulosta, verta on niin paljon, että pelin opetuksellisuus ei juuri tule ilmi. Päätin kokeilla versiota, joka vastaisi paremmin pelin tarkoitusta. Tein vain pienen muutoksen, mutta se näytti toimivan mainiosti.



Kuva 19. Logon lopullinen versio

Tässä kuvan 19 mukaisessa uudessa versiossa on vain verta itse sorvessa. Veri on niin hentoisesti siellä, että se ei mielestäni häiritse pelin opetuksellista

puolta. Samalla pelin nimi tulee paremmin esiin. Logon valmistuttua lisäsin sen näkyviin eri objekteihin Unity:n sisällä.

Logon jälkeen pääsin taas keskittymään visuaalisen sisällön luomiseen. Ryhmän muilta jäseniltä tuli 3D-mallien pyyntöjä, joita minä toteutin. Peliin tuli nyt mm. turvalasit, työkalupöytä, ja sorvin työkalujen osia. Aina kun olin saanut 3D-mallin mallinnettua alusta loppuun, testasin VR:ssä, miltä ne näyttäisivät. Useimpia ongelmia, mitä tuli vastaan, oli koon ja suhteiden kanssa. Usein valmista mallia jouduin jälkeinpäin korjaamaan, jotta ne näyttäisivät siltä, miltä pitäisi myös virtuaalitodellisuudessa. Oli otettava huomioon myös se, miltä objektit näyttivät kauempaa ja miltä ne näyttivät, kun niitä pideltiin kädessä. Työkalujen kanssa oivalsin uuden ongelman, jota VR-tekniikka toi tullessaan. Koska työkalua voi pyöritellä kädessä miten haluaa, niihin ei synny ns. sokeaa kulmaa, johon normaalisti voi laittaa tekstuurien saumat. Huomatessani tämän, oli minun muokattava valmiita tekstuureita objekteissa, joita sai käsissä pyöriteltä. Sain häivytettyä saumat, kun lisäsin molempiin tekstuurin rajoihin samaa sävyä sauman kohdalle.

Seuraava tehtäväni oli kentäsuunnittelu. Ryhmäni jäsenet olivat suunnitelleet pelimme niin, ettei pelaaja pysty liikkumaan paljoa sorvin edestä kauemmas. Tällä tavalla ei tulisi HTC Vive:n ongelmia niin helposti eteen, jossa pelaaja voi kompastua johtoihin tai että oikean maailman seinät tulisivat vastaan. Tässä tehtävässä sain käyttää täyttä päätäntävaltaani, eikä minun tarvinnut luoda erilaisia visuaalisia ehdotuksia ryhmäni jäsenille. Ensimmäiseksi minun oli luotava sorville ympäristö. Tämä on se huone, jossa sorvin käyttöä opetellaan virtuaalisorvin kanssa. Peliin tulee vielä toinen huone, joka toimii valikkona. Minun olisi sovitettava nämä kaksi huonetta niin, että ne sopivat keskenään. Koska sain täysin vapaat kädet, lähdin suoraan jalostamaan ideaa sorvi huoneesta, jonka olin saanut jo aiemmin. Aloitin tämän prosessin tekemällä nopean luonnoksen siitä, mitä olin ennalta ajatellut.





Kuva 20. Sorvihuoneen luonnos

Kuvan 20 luonnoksessa sorvi on keskellä robotin kanssa ja muu ympäristö on vain tilan tunnelman luomiseen. Luonnostelin tehtaan varastohuoneen kaltaisen tilan, jossa olisi hyllyjä, koneita sekä paljon irtotavaraa. En panostanut luonnokseen tämän enempää, sillä minulla oli oma tarkka visioni siitä, miltä huoneen tulisi näyttää. Tällä kertaa minun ei tarvinnut ymmärrettävästi selittää muille sitä, mitä olin suunnitellut. Rajoitteina huoneessa on tila, jossa pelaaja voi liikkua sekä sen on oltava sen verran lowpoly, ettei peli hidastele. Huoneen olisi myös toimittava muun visuaalisen sisällön kanssa ja tuettava sitä. Koska olin ainoa visuaalisen puolen tekijä ryhmässä, sain tätä huonetta varten hakea Unity asset-kaupasta haluamiani valmiita objekteja. Kaikki kolmannen osapuolen 3D-mallit, jotka jättäisin peliin on teksturoitava uudelleen, jotta ne sopivat pelin tyyliin. Kun sopivia varastoon liittyviä asetteja oli tarpeeksi, lähdin työstämään huonetta Unity:n näkymällä. Rajasin pelialueen hieman sorvia suuremmaksi, käyttäen aitoja ja muita esteitä. Tein rajaavista aidoista sen verran korkeita, ettei pelaajalle tulisi korkean paikan pelkoa tai tunnetta siitä, että virtuaaliaitaan voi kompastua.

Mallinsin huoneelle seinät, katon ja lattian ja tein niistä hyvin suuria, jotta huoneesta tulisi tilava. Kun koko näkymä oli rajattu, aloin lisäämään hakemiani asetteja. Huomasin, kuinka haastavaa on luoda ympäristöjä. Minulla ei juuri ole ollut kokemusta kenttä- tai ympäristösuunnittelusta, mutta pyrin tekemään huoneen mielenkiintoiseksi parhaan kykyni mukaan. Kun olin saanut huoneen luonnoksen valmiiksi, kokeilin miltä se näyttää virtuaalitodellisuudessa. Sain taas

huomata, kuinka väärässä koossa melkein kaikki objektit ovat. Samalla myös assetit, jotka olin tuonut näkymään saivat huonovointisuutta aikaiseksi. Niissä oli realistiset tekstuurit ja eivät sellaisenaan toimineet VR-lasien kanssa. Vaikka koin huonovointisuutta aina, kun testasin kenttääni, tein sitä silti hyvin usein, jotta sain skaalauksen ja sommitelmat näyttämään hyvältä pelin sisällä. Asseteista ei löytynyt kaikkia haluamiani 3D-malleja, joten tein kenttääni varten niitä lisää. Omat objektini mallinsin niin polygoni-tehokkaasti, kuin näyttävästi pystyin. Esimerkiksi tynnyrit, jotka mallinsin, olivat ilman kaikkia niitä pintoja, joita ei muutenkaan näkisi. Tynnyreistä puuttui polygonien "facet" eli pinnat niin takaa kuin, pohjastakin. Tein useita erilaisia objekteja samalla tavalla. Kun testasin Unity:n näkymää pyörittelemällä, sain huomata sen, että omat mallini olivatkin todella paljon hyödyllisempiä, valmiisiin asetteihin verraten. Omat mallini olivat näkymättömiä takaa, joka mahdollisti näkymän täyden näkymisen ilman esteitä. Asetteina hankitut objektit olivat näkyviä kaikista kulmista, joka teki kaukaa katsomisen vaikeammaksi.

Käytin paljon aikaa pelikentän tekemiseen ja testaukseen. Vaihdoin järjestystä ja tekstuureita, yrittäen sovittaa kaiken keskenään. Aina kun olin tehnyt pienen määrän muutoksia, katsoin miten asiat toimivat VR:ssä. Kirjasin kaikki virheet ja tarvittavat muutokset paperille ja lähdin korjaamaan niitä ja tekemään taas uutta. Jatkoin tätä prosessia samanlaisena monta kertaa. Yleisimpiä virheitä oli tekstuurin väärä sävy sekä objektien kokoluokka. Kun sorvihuoneessa oli tavarat hyvässä sommitelmassa ja suhteessa keskenään, loin omat tekstuurit kaikkien tuotujen asettien päälle.



Kuva 21. Sorvihuoneen Unity näkymä

Valitettavasti Ulsterin yliopistossa ei ollut mitään menetelmää, jolla kaapata VR-laseilla nähtyä näkymää. Se, miten sorvihuone näkyi Unity:ssä kuten kuvassa 21, ei lyönyt millään tavalla vertoja siitä, miten sen kokonaisuudessaan koki VR-laseilla. Kun pelaajana seisoo huoneen keskellä olevassa pelitilassa, ympäröivä huone tuntui suurelta ja avaralta. Tulee tunne siitä, kuin olisit todella tässä varastohuoneessa sisällä. Alituisella testaamisella olin saanut visuaalisen sisällön sopusuhtaiseksi keskenään, ja kaikki alkoi näyttämään siltä, kuin niiden kuuluisikin. Tässä vaiheessa sain myös pientä apua kokeneemmalta visuaaliselta suunnittelijalta. Ohjaajani oli kutsunut tämän ammattilaisen vieraaksi pari päiväksi, jotta hän voisi antaa vinkkejä aloittelevalle pelien visuaaliselle suunnittelijalle. Hänen avullaan pääsin säätämään valaistusta, efektejä, varjoja ja värisävyjä. Tämä muutti taas kaikki periaatteet, miltä peli näyttää VR-laseilla. Testasin eri valon värisävyjä, useita varjojen ja valojen asetuksia ja savu-efektejä. Tein kokeilut ensin Unity:ssä, sitten VR:ssä. Tämä oli erittäin vaikea prosessi minulle, sillä valojen ja efektien säätäminen oli erittäin tarkkaa. Pienellä muutoksella oli mahdollista muuttaa koko huoneen tunnelma ja joissain asetuksissa jo hyväksi todetut tekstuurit vilkkuivat VR-näkymässä. Sain taas huonovointisuutta näistä testeistä, mutta onneksi ongelmia ei ollut vielääkään viiveen kanssa.

Kun pelin päähuone oli tehty, oli valikkona toimivan huoneen aika. Sain taas vapaat kädet, joten lähdin suunnittelemaan tästä huoneesta tehtaan taukotilaa. Tällä kertaa tarvitsin paljon vähemmän erilaisia objekteja, joten mallinsin melkein kaikki mallit itse. Tällä tavalla sain myös kontrolloitua polygonimäärää sekä



helpottamaan Unity-näkymän tarkastelua, näkymättömien osien ansiosta. Tein samanlaista prosessia kuin sorvihuoneessa, lisäsin objekteja, tekstuureita ja skaalasin, sitten katsoin VR-laseilla, kirjoitin kaikki virheet ylös ja menin korjaamaan. Monta kertaa kun pääsin katsomaan näkymää VR:ssä, tuli sivun verran asioita, joita pitäisi muuttaa. Valaistus, skaalaus, värien sopivuus ja kaikki piti testata VR-laseilla erikseen. Monta kertaa vaikka Unity-näkymä oli mainio, VR-lasit päässä löysi niin paljon virheitä, että sivullinen korjauksia tuli taas lisää.



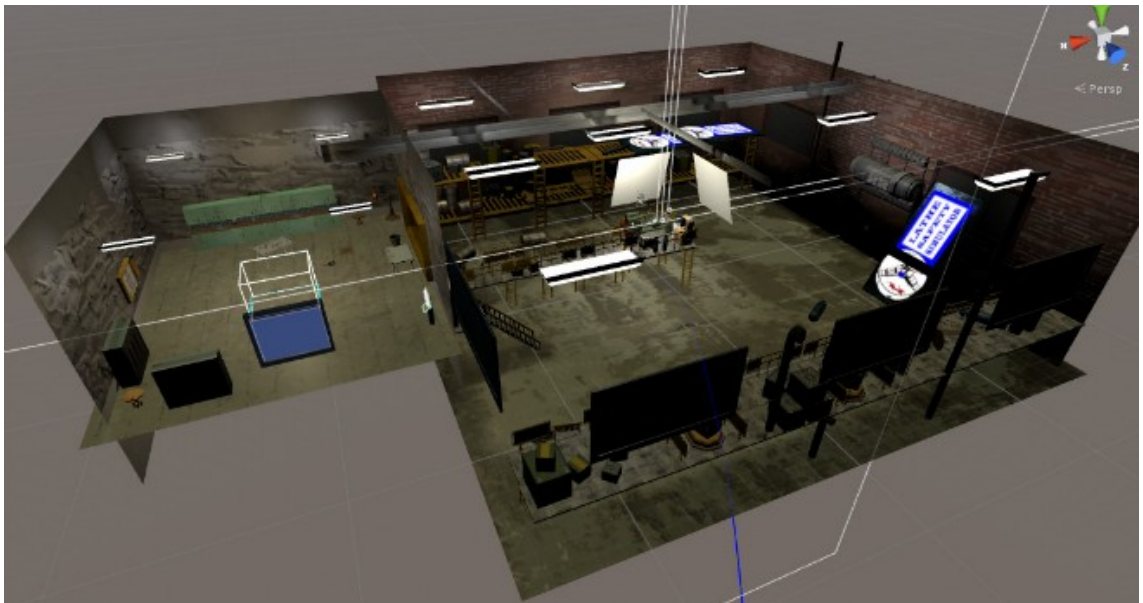
Kuva 22. Taukokuoneen Unity näkymä

Kuvan 22 taukokuoneen tekeminen oli yllättävän hauskaa, sillä tähän huoneeseen sain luoda hyvinkin tuttuja objekteja. Tein mm. kahvin- ja vedenkeitin, donitseja ja pullia. Yksi objekti oli tosin jätettävä tekemättä. Virtuaalituoli voisi saada pelaajan kaatumaan, jos pelaaja hetkeksi unohtaisi olevansa pelissä ja yrittäessä istua tyhjän päälle. Tämän huomattuani katsoin sorvihuoneenkin sillä silmällä, ettei mikään siellä kannustaisi pelaajaa istumaan tai nojaamaan virtuaaliobjektiin.

Pitkän testaus ja korjaus prosessin jälkeen huoneen tunnelmasta tuli haluttu. Kun taukokuoneessa kulkee, tulee hieman ankea ja likainen näkymä eteen, jossa kuitenkin on hieman kirkkaita värejä donitseissa. Tulee tunne siitä, että tänne on tultu töihin eikä taukoilemaan. Eritoten taukokuoneen pöydälle tein erilaisia objekteja, joita voisi olla hauska heitellä ympäriinsä, kuten edellä mainittuja donitseja ja pullia. Kun enemmän erilaisia ihmisiä pääsi tässä vaiheessa testaamaan taukokuonetta, huomasin sen, että on tärkeää, että huoneesta löytyy pal-

jon eri tavaroita tutkittavaksi ja heitettäväksi. Yleisin reaktio, mitä uusilla pelaajilla oli se, että katselivat huonetta, ottivat donitsista kiinni ja alkoivat viskomaan niitä ympäri huonetta. Tästä huoneesta olisi siis löydyttävä paljon heiteltävää.

Taukuhuoneen tekeminen toimivaksi oli loppujen lopuksi paljon vaikeampaa, kuin sorvi huoneen. Koska taukuhuoneessa kaikkialle on mahdollista mennä, oli oltava tarkkana siitä, että tavarat ovat oikeasti oikean kokoisia. Myös kaikki objektit oli oltava toimivia ja hyvän näköisiä myös joka puolelta katsottuna. Varsinkin ne tavarat, joiden kanssa pystyi interaktiivisesti kanssakäymään tai heittämään. Esimerkiksi donitsi mallieni alapuolella oli aluksi näkyvissä tekstuuri mapin sauma, joka oli mahdollista nähdä vain, jos otti donitsin käteen ja katsoi sitä alapuolelta. Normaalissa pelissä tällaista ongelmaa ei usein tulisi, sillä vaikka tavarat saattaisivat kaatua, ei niitä pääse yksityiskohtaisesti tutkimaan läheltä. Sain testata kaikki tekemäni objektit uudestaan niin, että tarkastin kaikki tekstuurien saumakohdat.



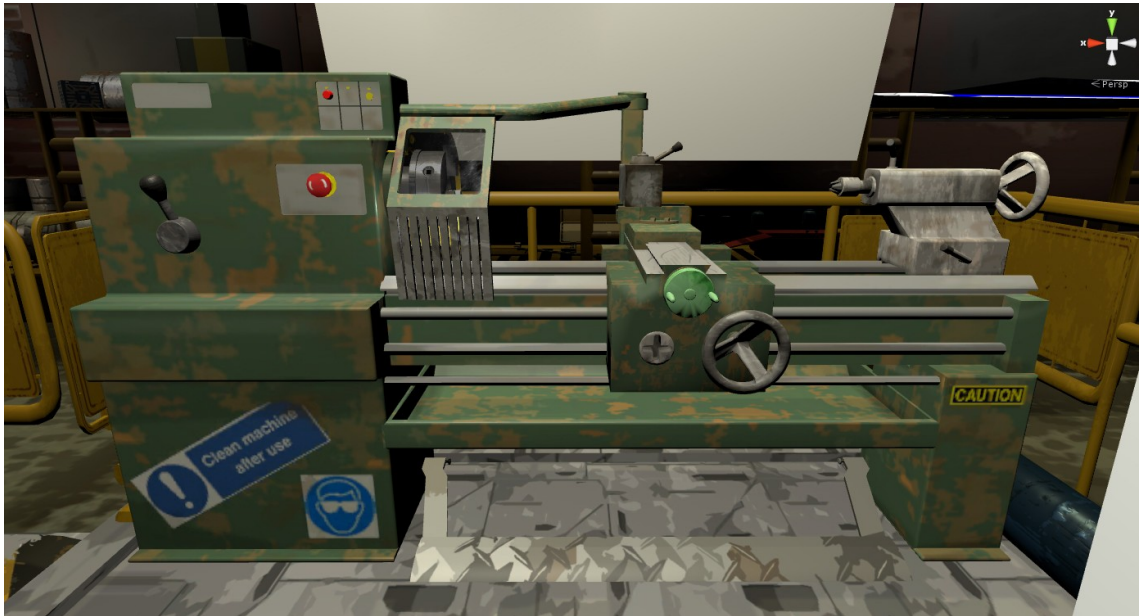
Kuva 23. Koko pelikenttä Unity näkymässä

Kun molemmat huoneet olivat suurimmalta osin valmiita, tein paljon töitä sen eteen, että ne toimisivat hyvin myös keskenään. Kuvasta 23 on keskellä nähtävissä taukuhuoneen ikkuna, josta pelaaja näkee suoraan sorvihuoneeseen ja toisinpäin. Sain käyttöni BitMap2Material ohjelmiston, jolla sain lisää todentuntua 3D-malleihini. Mutta äkkiä sain huomata sen, että bump-map ei toimi VR-laseilla ollenkaan. Kun stereonäöllä näkee objektin, jossa on bump-map, se näyttää vain siltä, että mallin päälle olisi huonosti piirrettyjä tummia ja vaaleita koh-

tia. Syvyyden illuusiota ei päässyt tapahtumaan, mikä taas aina toimii normaaleissa yhden ruudun peleissä. Onneksi huomasin tämän ongelman hyvinkin nopeasti, joten sain poistettua objekteista tekemäni bump-mapit. Sain myös huomata, että kun lisäsin specular-mappeja joihinkin tavaroihin, tuloksista tuli ristiriitaisia. Välillä specular-efektit eivät toimineet ollenkaan, mutta välillä ne saivat objektin näyttämään todella hyvältä. Kun tutkin, mistä oli kyse, niin sain huomata, että jos sileällä pinnalla on kiiltoa, se näyttää hyvältä VR:ssä. Mutta jos rosiossa pinnassa on kiiltoa, kuten esimerkiksi taukahuoneeni kivi seinässä, nämä kiillot vilkkuvat VR-näkymässä. Tämä vilkkuminen oli vielä häiritsevämpää, kuin raidat tekstuureissa ja laukaisivat vahvan huonovointisuuden aallon.

Kun koetin Bitmap2Material-ohjelmalla saada lisää realistisuutta peliin, tein aluksi paljon virheitä, joista opin, mitä realistisuuden piirteitä voi tuoda ja mitä ei. Missä tilanteissa ne toimivat, ja missä ne taas jopa tuhoavat pelikokemuksen laukaisemalla huonovointisuuden. Tämä riippui paljon yksittäisistä elementeistä, niiden ympäristöstä ja valaistuksen suhteesta. Mitä enemmän vain käytin aikaa testaamiseen, sitä paremmalta sain pelin näyttämään kokonaisuudessaan. Meni vielä monta tuntia siinä, kun hioin pelin yksityiskohtia. Valaistuksen, värisävyjen ja muiden muutosten jälkeen oli aina paljon korjattavaa, kun testasi niitä itse VR-näkymässä. Tässä prosessissa sain huomata sen, että kiilto toimi erinomaisesti sileässä pinnassa ja bump-mappeja oli rajoitetusti mahdollista käyttää. Bump-mapit näyttivät toimivan vain, jos kyseisen objektin lähelle tai ympäri ei pääse. Esimerkiksi taukuhuoneen kattoon sain sopimaan bump-mapin, ilman että se näytti huonolta. Mutta lattiaan taas ei voinut lisätä bump-mappia, sillä lattiaa oli mahdollista päästä todella lähelle. Sama sääntö toimi myös seinissä.

Tämä viimeinen hiominen oli erittäin työläs ja koetteleva prosessi, sillä väärin tehty usein tarkoitti huonovointisuutta. Koska opin parhaiten tekemistäni virheistä, se käytännössä tarkoitti sitä, että huonovointisuutta sain kokea pitkin päivää. Onneksi olin sentään heti alussa huomannut sen, että pelin on toimittava moitteettomasti, jotta tasapaino ei horju. Tämän takia olin koko pelin rakentanut sillä tavalla, että siinä olisi mahdollisimman vähän polygoneja.



Kuva 24. Valmis teksturoitu sorvi ympäristössään

Kun olin saanut realistisuuden niin korkealle kuin VR:ssä oli mahdollista, pelistä tuli visuaalisesti erittäin laadukas. Pelin pääelementti eli kuvassa 24 näkyvä sorvi näytti luonnolliselta VR-lasien näkymässä. Sain jopa materiaalin tuntua ympäristöön, kun rajatusti pystyin käyttämään esimerkiksi bump- ja specular-mappeja. Sain sorviin metallin, muovin sekä lasin tuntua ilman, että mikään näistä ominaisuuksista häiritsi VR:ssä. Mutta tämä oli ollut erittäin vaativa prosessi, sillä jokainen säätö oli tarkasti harkittava ja testattava erikseen VR-laseilla.

Tämä kaikki tuotti tulosta, sillä pelin visuaalinen ilme, minkä olin saanut aikaan oli erittäin yhteneväinen, tarkoituksenmukainen sekä näyttävä. Visuaalisen sisällön luomisprosessi oli ollut välillä erittäin haastavaa ja epämukavaa, mutta se oli sen arvoista, koska sain nähdä, mitä tällä kaikella vaivalla sain aikaiseksi. Tämä peli oli ensimmäinen oikea pelini ja samalla ensimmäinen kerta, kun pääsin testaamaan VR-teknologiaa sekä visuaalisensisällön tuottamista tähän tekniikkaan. Nämä huomioon ottaen, en voisi olla tyytyväisempi lopputulokseen, jonka sain ainoana visuaalisen puolen tekijänä aikaiseksi.

## 5 JOHTOPÄÄTÖKSET

Tutkimuksestani kävi ilmi monta visuaalista seikkaa, jossa monikin ratkaisu yleensä toimi ja myös ratkaisuja, jotka eivät toimineet muissa kuin poikkeustapauksissa. Jos olisi ollut enemmän aikaa tutkimukseen ja testaukseen, olisin varmasti löytänyt lisää näitä poikkeuksellisia tilanteita, joissa virtuaalitodellisuuden visuaalisesti hankalat osa-alueet olisivat toimineet.

Yhteenvedona visuaaliseen suunnitteluun vaikuttavat seikat, kun sisältöä luodaan virtuaalitodellisuuspeliin:

Pelin on toimittava saumattomasti, jotta pelaajan tasapaino ei järky eikä hän koki pahoinvointia. Kuvitteellisen horisontin on myös pysyttävä samassa paikassa, tai pelaaja voi kaatua. Ongelmatilanteissa, kuten pelin kaaduttua, tämä on myös otettava huomioon.

Pelialue on suunniteltava niin, että huomioi oikean maailman seinien tuomat rajoitteet. Jos pelaajan on matkustettava pitkiä matkoja, teleporttaus on suositeltavaa. Pelialueen suunnittelussa on otettava huomioon myös VR-laseista roikkuvat johdot sekä esteiden aiheuttama, suunnan laskemisen estyminen ohjaimissa sekä laseissa. Pelialueen suunnittelussa on myös otettava huomioon pelaajan tuntemukset, jos esimerkiksi kaiteita, tuoleja ja muita objekteja on pelissä. Pelaajaa ei saa rohkaista istumaan virtuaalituoliin, nojaamaan kaiteeseen tai muuhunkaan tukeen, joka voi kaataa pelaajan. On myös hyvä ottaa huomioon se, että jos pelialueeksi on mallinnettu korkealla kielekkeellä oleva tila, kaiteiden olisi oltava sen verran korkeat, että pelaaja ei koe helposti horjahtavansa. Korkeat kaiteet antavat visuaalista suojaa ja estävät pelaajaa menemästä pelialueen reunalta alas. On myös hyvä ottaa huomioon esimerkiksi akrofobia, jossa pelaaja voi saada paniikin omaisen pelon korkeista paikoista.

Mitä latteammin ja yksinkertaisemmilla tekstuureilla pelin visuaalinen puoli on suunniteltu, sitä helpommin on saavutettavissa usean eri pelaajan pelaaminen ilman pahoinvointia. Pahimman reaktion suuressa osassa pelaajia on mahdollista saada aikaan kapeilla linjoilla tai raidoilla, jotka vilkkuvat ja värisevät VR-laseja käytettäessä. Tietyt väriyhdistelmät, kuten esimerkiksi korkeat värikontrastit, voivat saada myös samankaltaisen reaktion aikaan. Parhaan tuloksen teks-

tuurien visuaalisessa suunnittelussa on mahdollista saada aikaan vain testamalla eri variaatioita virtuaalitodellisuudessa ja mieluiten monella eri pelaajalla.

Erikoistekstuurit, kuten bump-, specular- ja normal-map, toimivat erittäin vaihtelevasti virtuaalitodellisuudessa. Näitä efektejä käytettäessä on otettava huomioon stereonäön ominaisuus, jossa pelin objektit on mahdollista nähdä ikään kuin kahdelta ruudulta samaan aikaan hieman eri kohdissa. Tämä tekee eritoten bump-mappien käytöstä erittäin haastavaa, mutta kaukana olevissa kohteissa iluusio vielä toimii. Specular-map toimii parhaiten sileissä pinnoissa, joissa ei tule raitaefektiä, joka saa näkymän vilkkumaan. Kaikkien erikois-mappien kanssa on oltava varovainen ja niiden toimivuus on testattava yksitellen.

Virtuaalitodellisuuspelissä on vielä otettava huomioon pelaajan kunto. Pelissä olisi vältettävä toistuvaa räsytystä, kuten esimerkiksi lattialta alituisen nostettavia tavaroita. Tämä ongelma on helppo välttää järkevällä pelisuunnittelulla.



## 6 POHDINTA

Parasta tässä projektissa oli se, että sain keskittyä täysin visuaaliseen sisällön luomiseen. Minun ei tarvinnut miettiä mitään koodista tai toiminnoista, vaan sain keskittyä täysin pelin visuaaliseen puoleen. Hallitsin 3D-mallintamisen entuudestaan, mutta olin täysin kokematon VR-lasien ja koko teknologian kanssa. Koska osasin oman osa-alueeni hyvin, sain käyttää kaikki ylimääräiset resurssini uuden teknologian ymmärtämiseen ja testaamiseen. VR-pelit tulevat olemaan varmasti kokemuksen arvoista kaikille pelien ystäville, varsinkin kun tekniikka vielä kehittyy eteenpäin. Pelaaminen pelin sisällä on kokemukseltaan niin ainutlaatuinen ja säväyttävä, että kerran kokeiltuaan sitä pyrkii kokemaan uudelleen. Mitä erilaisia pelejä, maailmoja tai simulaatioita VR-tekniikka tuokaan tulevaisuudessa, en voi muuta kuin toivoa, että saan olla tämän mullistavan tekniikan kehityksessä mukana jatkossakin.

Se määrä, mitä kaikkea opin virtuaalitodellisuuspelin tekemisestä oli huomattava. Tietämättä ennestään melkein mitään siitä, mitä virtuaalitodellisuus tarkoittaa, tiedän nyt paljon VR:n ominaisuuksista, mahdollisuuksista sekä siihen visuaalisen sisällön tuottamisesta. Sisällön tuottamisessa aivan kaikki visuaalinen materiaali, joka peliin laitetaan, on katsottava VR-laseilla erikseen, ennen kuin siitä voi tietää, toimiiko se. Mitä enemmän testejä tein, sitä paremmaksi sain pelini visuaalisesti. Tämä on vaikea prosessi, sillä huonovointisuuden kokemus on osa testausprosessia. Mutta tämä oli tehtävä, sillä minulla ei ollut vielä mahdollisuutta ohjeistukseen visuaaliseen suunniteluun, virtuaalitodellisuutta varten.

Opin tästä kaikesta sen, miksi The Lab ja Job Simulator ovat valinneet tyykinsä. Molemmissa peleissä ei tule huonovointisuutta, sillä peleissä ei ole kiiltoja eikä muita ongelmallisia erikoistekstuureita. Pinnat ovat hyvin yksivärisiä ja sileitä, johon päädyin myös itse monen tekemäni testin jälkeen.

Nopeasti opin sen, että pelin on oltava sujuva, jotta tasapaino ei järky. Mutta kaiken muun oppimiseen meni monta tuntia testien tekemistä. Toivon, että opinäytteeni olisi sen verran hyödyllinen seuraavalle visuaaliselle suunnittelijalle, jotta heidän ei tarvitse käydä samoja testejä enää läpi. Olen tuonut esiin paljon virheitä, jotka seuraavan on helppo välttää.

## LÄHTEET

Autodesk 2016. Animation. Viitattu 6.2.2017

Autodesk 2017. About Exporting Model Data. Viitattu 17.1.2017

Birn, J. 2002. 3D Rendering. Viitattu 11.4.2017 <http://www.3drender.com/glossary/3drendering.htm>

Ehrenhaus, S. 2014. Rigging Guideline for the Artist: What's Important for a Good Rig? Viitattu 16.4.2017 <http://blog.digitaltutors.com/rigging-guideline-artist-whats-important-good-rig/>

Freescape Semiconductor 2010. 3D Animation Techniques on the i.MX31 PDK. Viitattu 14.2.2017 <http://www.nxp.com/assets/documents/data/en/application-notes/AN4044.pdf>

Groningen Yliopisto. 2015. Virtual Reality and Visualization. Viitattu 18.1.2017 [http://www.rug.nl/society-business/centre-for-information-technology/research/hpcv/vr\\_visualisation/?lang=en](http://www.rug.nl/society-business/centre-for-information-technology/research/hpcv/vr_visualisation/?lang=en)

Historia 2012. Historianet. Milloin 3d-kuvat keksittiin? Viitattu 27.1.2017 <http://historianet.fi/tekniikka/keksinnot/milloin-3-d-kuvat-keksittiin>

Koskenlaakso, L. 2016. Kolmiulotteinen virtuaalitodellisuus ja lisätty todellisuus – Kohti uusia maailmoja. VTT. Viitattu 16.2.2017 <http://www.vtt.fi/Impulssi/Pages/Kolmiulotteinen-virtuaalitodellisuus-ja-lis%C3%A4tty-todellisuus-Kohti-uusia-maailmoja.aspx>

Masters, M. 2014a. Key 3D Rigging Terms to Get You Moving. Viitattu 16.1.2017 <https://www.pluralsight.com/blog/film-games/key-rigging-terms-get-moving>

Masters, M. 2014b. Key Terminology for Mastering 3D Animation. Viitattu 16.1.2017 <http://blog.digitaltutors.com/common-terminology-3d-animation/>

PC. 2016. Enkylopedia. Definition of 3D-Animation. PCMag Digital Group. Viitattu 16.1.2017.

Proteus 2015. Steam VR Template. Viitattu 6.2.2017 <https://forums.unrealengine.com/showthread.php?106609-Steam-VR-Template>

Pänkäläinen, T. 2016. Virtuaalitodellisuus – 120 miljardin markkina vuonna 2020? Viitattu 18.1.2017 <http://www.virtuaalimaailma.fi/virtuaalitodellisuus/>

Silverman, D. 2013. 3D Primer for Game Developers: An overview of 3D Modeling in Games. Viitattu 16.01.2017 <http://gamedevelopment.tutsplus.com/articles/3d-primer-for-game-developers-an-overview-of-3d-modeling-in-games--game-dev-5704>

Slick, J. 2016. What is Rigging? Viitattu 16.1.2017 <https://www.lifewire.com/what-is-rigging-2095>



Simmons, M., Wilhelms, J. & Van Gelder, A. 2002. Model-based Reconstruction for Creature Animation. University of California. Santa Cruz

Tumbokon, K. 2016. How To Set Up PlayStation VR. Viitattu 6.2.2017

Tuominen, A. 2016. Työharjoittelupäiväkirja

Unity 2017a. Animator Scripting. Viitattu 6.2.2017

Unity 2017b. Asset Workflow. Viitattu 11.4.2017 <https://docs.unity3d.com/Manual/AssetWorkflow.html>

Unity 2017c. FBX Importer, Rig options. Viitattu 16.2.2017 <https://docs.unity3d.com/Manual/FBXImporter-Rig.html>

Unity 2017d. The leading global game industry software. Viitattu 24.1.2017 <https://unity3d.com/public-relations>

Unreal Engine 2017. What is Unreal Engine 4? Viitattu 24.1.2017 <https://www.unrealengine.com/what-is-unreal-engine-4>

Vive 2017. Choosing the play area. Viitattu 6.2.2017 [https://www.vive.com/us/support/category\\_howto/720449.html](https://www.vive.com/us/support/category_howto/720449.html)

VRS 2016a. Virtual Reality Society. History Of Virtual Reality. Viitattu 27.1.2017 <http://www.vrs.org.uk/virtual-reality/history.html>

VRS 2016b. Virtual Reality Society. What is Virtual Reality? Viitattu 18.1.2017 <http://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html>

Wang, H. 2016. Texture Mapping. Ohio State University.

Wepopedia. 2017. Export. Viitattu 17.1.2017 <http://www.webopedia.com/TERM/E/export.html>

Wikipedia 2017a. Kolmiulotteinen mallintaminen. Viitattu 16.1.2017 [https://fi.wikipedia.org/wiki/Kolmiulotteinen\\_mallinnus](https://fi.wikipedia.org/wiki/Kolmiulotteinen_mallinnus)

Wikipedia 2017b. Liikkeenkaappaus. Viitattu 16.1.2017

Wikipedia 2017c. Motion capture. Viitattu 16.1.2017 [https://en.wikipedia.org/wiki/Motion\\_capture](https://en.wikipedia.org/wiki/Motion_capture)

Wikipedia 2017d. Teksturointi. Viitattu 16.1.2017 <https://fi.wikipedia.org/wiki/Teksturointi>